

Thesis (Spring 2013)

Report on

***A Wireless Controlled semi-autonomous Fire
Extinguishing Robot to rescue and monitor hazardous
place***



Thesis Report Submitted To: Dr. Mohammed Belal Hossain Bhuian
Assistant Professor, Department of EEE
BRAC University.

Thesis Group Members:

- | | | |
|-----|-----------------|----------|
| (1) | Imran Bin Jafar | 10121017 |
| (2) | Kanij Raihana | 10121032 |
| (3) | Tanwy Barua | 10121064 |
| (4) | Milon Uddin | 10121034 |

Thesis Supervisor: Dr. Mohammed Belal Hossain Bhuian

Thesis Co- Supervisor: Mr. Risul Karim (Lecturer-I)

Table of Contents

Declaration.....	4
Acknowledgement.....	4
Abstract.....	1
Chapter 1: Introduction	14
1.1 Introduction	15
1.2 Worldwide Statistical Report	16
1.3 History of Accident due to fire in Bangladesh	17
1.4 What are robots	18
1.5 Importance of unmanned fire extinguisher robot	18
Chapter 2: Design & Construction Overview	20
2.1 Concept of Fire Extinguishing Robot.....	21
2.1.1 System Architecture	22
2.2 Mechanical Design	23
2.2.1 Logical aspects Behind Structural Designing.....	23
2.2.2 CAD (Computer aided drawing) Model.....	28
2.2.3 Measurement of Robot	29
2.3 Materials for robotic body	30
2.4 Primary Equipments	37
2.4.1 Microcontroller Board	37
2.4.1.1 Introduction to Arduino ADK	37
2.4.1.2 Introduction to Arduino Mega2560.....	39
2.4.2 Optocoupler	42
2.4.2.1 Working Principle	42
2.4.2.2 Purpose of 4N35 in this project.....	43
2.4.2.2 Experimental Analysis.....	44
2.4.3 Power MOSFET	44
2.4.3.1 IRFP250N	44
2.4.3.2 IRFP250N Characterisrics.....	45

2.4.4 Relay	46
Chapter 3: Study of Motors for FireBOT	49
3.1 Motor	50
3.1.1 Motor torque, RPM and Power calculation	51
3.1.2 Experiment with motor	58
3.2 Motor Controlling circuit	64
3.2.1 Condition of a Single Motor for Each Logic Combination	66
3.2.1.1 Logic Combination 00	66
3.2.1.2 Logic Combination 01	68
3.2.1.3 Logic Combination 10	69
3.2.1.4 Logic Combination 11	71
3.2.2 Fire Extinguisher Robot Motor Controller Operation	73
Chapter 4. Communication System for FireBOT	74
4.1 Communication	75
4.1.1 Serial Communication Between Computer & Arduino:	75
4.1.2 Basic Arduino Coding for Serial Communication:	77
4.2 nRF24L01 + LNA + PA Transceiver Communication	77
4.2.1 Serial Communication Between Computer & Arduino:	79
4.3 Functional Block Diagrams of NRF24L01	80
4.3.1 PA - (Power Amplifier)	82
4.3.2 Transmission (TX) Filter	82
4.3.3 GPSK modulator	83
4.3.4 TX FIFO	83
4.3.5 LNA - Low Noise Amplifier	84
4.3.6 RF Synthesizer	84
4.4 Details of nRF24L01 Wireless Module	85
4.4.1 Radio Control	86
4.4.2 State Diagram	86
4.4.3 Operational Modes of nRF24L01	88
4.4.3.1 Power DOWn Modes	88
4.4.3.2 Standby Mode	88

4.4.3.3 Receiving Mode.....	89
4.4.3.4 Transmission Mode	89
4.5 Data Tempo of nRF24L01	89
4.6 RF Channel Frequency	90
4.7 PA (Power Amplification) Control.....	90
4.8 RX/TX Control	90
4.9 Enhanced ShockBurst	91
4.9.1 Enhanced ShockBurst Overview	91
4.10 Arduino Mega and nRF24L01 Communication Process	91
4.10.1 Serial Peripheral Interface	92
4.11 Experimental Data Analysis of nRF24L01 with FireBOT	95
Chapter 5: Fire Extinguishing System Design	101
5.1 Extinguishing Unit	102
5.1.1 Fire Extinguisher	102
5.1.2 Solenoid valve	106
5.1.2.1 Mechanism of Solenoid valve	106
5.1.2.2 Use of Solenoid valve in Extinguisher	108
5.1.2.3 Control Circuit for Solenoid Valve	110
5.1.3 Rotation of Extinguisher nozzle	112
Chapter 6: Sensor System Design.....	113
6.1 IP Camera	114
6.1.1 Specifications	116
6.1.2 Communication Techniques with Fire Fighting Robot	116
6.2 Interface with sensors	117
6.2.1 Temperature Sensor	117
6.2.1.1 General Description of LM35	118
6.2.1.2 Temperature Measurement	120
6.2.2 Ultrasonic Sensor	121
6.2.2.1 Theory of Operation	122
6.2.2.2 Distance Measurement	123
6.2.2.3 Experimental Measurement	124

6.2.3 PIR (Detection of human)	127
6.2.3.1 Working Principle	128
6.2.3.2 Effects of Atmosphere	129
6.2.3.3 PIR sensor in Flame Detection	130
Chapter 7: Portable Power Supply Design.....	131
7.1 Portable Power Supply	132
7.1.1 For IP Camera	132
7.1.2 For Motor & Arduino	136
Chapter 8: Software Design	137
8.1 Software Implementation	138
8.1.1 Introduction to Microsoft Visual Studio 2010	138
8.1.2 Control Unit Software	138
8.1.3 Features of Control Unit Software	139
8.1.4 Functions of Control Unit Software	139
8.2 Block Diagram of Software.....	140
8.2.1 Block Diagram of Software Interface with Hardware.....	142
8.3 Software Manual	142
Chapter 9: Conclusion.....	147
9.1 Summary	148
9.2 Final Remark	149
9.3 Future Plan	149
References	151
Appendix.....	155

List of Figures:

Figure 1.1: Worldwide statistical inspection of casualties due to fire between (1845-2012)	1
Figure 1.2: Bangladeshi statistical inspection of casualties due to fire	4
Figure 2.1: Illustration of Hardware Design	1
Figure 2.2: Illustration of Software Design	4
Figure 2.3: Probable Chassis Design	1
Figure 2.4: Upper Chamber of Fire Extinguisher Robot	4
Figure 2.5: Lower Chamber of Fire Extinguisher Robot	1
Figure 2.6: Top view of robot design	4
Figure 2.7: Simplified 2D model of robot with fire extinguisher	1
Figure 2.8: 3D model of Extinguisher robot in AutoCAD frame	4
Figure 2.9: Practical Structure of Fire Extinguisher Robot	1
Figure 2.10: Volume per unit weight of Aluminum, steel, copper, Brass	4
Figure 2.11: Copper Wire	1
Figure 2.12: Bronze bearing	4
Figure 2.13: Thermal conductivity of Aluminum Compared with other Metals	1
Figure 2.14: Aluminum Wheel base	4
Figure 2.15: steel frame for robot	1
Figure 2.16: Arduino ADK board.....	4
Figure 2.17: Arduino Mega 2560	1
Figure 2.18: Internal Structure of Optocoupler (4N35)	4
Figure 2.19: Working principle of Optocoupler	1
Figure 2.20: Connection diagram of 4N35 with Arduino Mega in ISIS 7 Professional	4
Figure 2.21: IRFP250N MOSFET	1
Figure 2.22: Basic circuit diagram for driving a dual pole relay	4
Figure 2.23: Experimenting with Dual pole Relay and Arduino Mega.....	1
Figure 3.1: acceleration a in a horizontal floor	4
Figure 3.2: Climbing 30° Slope with acceleration	1
Figure 3.3: horizontal floor in a constant velocity	4
Figure 3.4: Time vs Voltage curve at constant Current, $I = 1.2A$	1

Figure 3.5: Time vs Voltage curve at constant Current, $I = 1.3A$	4
Figure 3.6: Time vs Voltage curve at constant Current, $I = 1.4A$	1
Figure 3.7: Time vs Voltage curve at constant Current, $I = 1.5A$	4
Figure 3.8: Time vs Voltage curve at constant Current, $I = 1.6A$	4
Figure 3.9: Time vs Voltage curve at constant Current, $I = 1.7A$	1
Figure 3.10: Time vs Voltage curve at maximum current	4
Figure 3.11: Current vs Voltage curve	1
Figure 3.12: Current vs RPM curve	4
Figure 3.13: Motor Controlling Circuit	1
Figure 3.14: Motor stop for logic 00	4
Figure 3.15: Motor rotate clockwise for logic 01	4
Figure 3.16: Motor stop for logic 10	1
Figure 3.17: Motor rotate anti-clockwise for logic 11	4
Figure 4.1: Arduino is operated in 2 communication schemes	1
Figure 4.2: Serial Monitor: Monitoring and Setting Up serial Communication Baud Rate	4
Figure 4.3: nRF24L01 Module in the unconnected state	1
Figure 4.4: nRF24L01 is connected with Arduino (ATmega 2560 Microcontroller)	4
Figure 4.5: General Wireless Communication among Access point and clients	1
Figure 4.6: nRF24L01+ module's internal circuitry block diagram	4
Figure 4.7: Expansion of nRF24L01+ Wireless Module's major Three electronic components ...	1
Figure 4.8: nRF24L01+ Wireless Module's other Four important electronic components	4
Figure 4.9: Digital Filtering block diagram	1
Figure 4.10: RF synthesizer Block diagram	4
Figure 4.11: An example of a state diagram	1
Figure 4.12: State diagram of nRF24L01+ Module	4
Figure 4.13: RF output power setting for the nRF24L01+	1
Figure 4.14: Packet format of nRF24L01+ with MSB to the left.	4
Figure 4.15: Pin connection of nRF24L01+ Wireless Module	1
Figure 4.16: SPI Master Device with a single SPI Slave Device	4
Figure 4.17: SPI Master is connected with multiple SPI slave devices	1
Figure 4.18: Laboratory setup and controlling RF Communication System	4

Figure 4.19: Signal at CE pin while Transmission	1
Figure 4.20: Signal at CSN pin while Transmission	4
Figure 4.21: Signal at MOSI pin while Transmission	1
Figure 4.22: Perfect Signal at SCK pin while Transmission	4
Figure 4.23: Signal at MISO pin while Transmission	1
Figure 4.25: Signal at MISO (Yellow) and MOSI (Blue) in Reception Mode	4
Figure 5.1: (a) Different Parts of a fire extinguisher; (b) A stored-pressure fire extinguisher	1
Figure 5.2: water extinguisher	4
Figure 5.3: Dry powder extinguisher	1
Figure 5.4: Foam extinguisher	4
Figure 5.5: CO_2 extinguisher	1
Figure 5.6: Design of a basic solenoid valve	4
Figure 5.7: Different parts of a Solenoid Valve	1
Figure 5.8: Solenoid Valve Control Circuit Simulation (Solenoid valve turned on)	4
Figure 5.9: Solenoid Valve Control Circuit Simulation (Solenoid valve turned off)	1
Figure 6.1: Wireless day/ night surveillance camera TL-SC3171G	1
Figure 6.2: Linearity of thermistor and temperature sensor	4
Figure 6.3: Circuit connection of LM35	1
Figure 6.4: Pin diagram of LM35	4
Figure 6.5: Pin diagram of TS-601	1
Figure 6.6: Distance measurement with the help of microcontroller.....	4
Figure 6.7: Theoretical pulses emission and receives through ultrasonic sensor	1
Figure 6.8: Circuit connection for Sonar Sensor	4
Figure 6.9: Oscilloscope observation of sonar sensor	1
Figure 6.10: Graph between actual distance and measured distance.....	4
Figure 6.11: Parallax PIR sensor	1
Figure 6.12: Pin diagram of Parallax PIR sensor.....	4
Figure 6.13: Internal structure of PIR sensor.....	1
Figure 6.14: PIR sensor operation by sensing different infrared radiation.....	4
Figure 6.14: Effects of temperature on PIR Sensor	1

Figure 7.1: Pin Configuration of LM7812	1
Figure 7.2: Internal circuitry of LM7812	4
Figure 7.3: Connection of LM7812	1
Figure 7.4: Practical Implemented Circuit	4
Figure 7.5: Input vs Output Voltage of LM7812	1
Figure 7.6: Input vs Output current of LM7812	4
Figure 8.1: Control Unit software	1
Figure 8.2: Block Diagram for Sending Keyboard Command	4
Figure 8.3: Block Diagram of Receiving Sensor data from Serial Port	4
Figure 8.4: Block Diagram of Software and hardware Interface	4
Figure 8.5: User Interface of Firebot	4
Figure 8.6: Identifying COM Port	4
Figure 8.7: Successful Connection of COM Port	4
Figure 8.8: Failed in connecting with COM Port	4
Figure 8.9: Sending “W” command	4
Figure 8.10: Generating Sensor data in Data Grid	4

List of Tables

Table 2.1: Estimate weight of different parts of robot	1
Table 2.2: Characteristic comparison of different materials	4
Table 2.3: Effects for Radiant Heat	1
Table 3.1: Calculation of torque for different force.....	1
Table 3.2: Modified torque calculation for different force	4
Table 3.3: Experimental result of geared DC motor	1
Table 3.4: Logic Table for Motor Control	4
Table 3.5: Function Table for Fire Extinguisher Robot Control	1
Table 4.1: Functions of 7 pins of NRF24L01+	1
Table 5.1: Types of extinguisher, agent, and class and sample application_	1
Table 6.1: Specifications of TL-SC3171G IP Camera	1
Table 6.2: Description of LM35 Pin_	4
Table 6.3: Pin description of TS-601	1
Table 6.4: Experimental result from TS-601 sensor	4
Table 6.5: Pin description of PIR sensor	1
Table 8.1: Keyboard Functions in Firebot Software	4

Declaration

We do hereby declare that the thesis entitled “A Wireless Controlled semi-autonomous Fire Extinguishing Robot to rescue and monitor hazardous place” is submitted to the Department of Electrical and Electronic Engineering of BRAC University in partial fulfillment of the Bachelor of Science in Electrical and Electronic Engineering. This is our original work and was not submitted elsewhere for the award of any other degree or any other publication.

Date:

Dr. Md. Belal Hossain Bhuian

Thesis Supervisor

Risul Karim

Thesis Co-supervisor

Imran Bin Jafar

ID: 10121017

Kanij Raihana

ID: 10121032

Tanwy Barua

ID: 10121064

Milon Uddin

ID: 10121034

Acknowledgement

First of all we are thankful to Almighty Allah for his blessing upon us and bestowing us courage to go with such task. This thesis would not have been possible without all the help and support we have received. We would like to thank our thesis supervisor Dr. Md. Belal Hossain Bhuiyan, Assistant Professor, Department of Electrical and Electronic Engineering, BRAC University and thesis co-supervisor Risul karim, Lecturer, Department of Computer Science and Engineering, BRAC University for guiding us throughout our thesis work. Special thanks for helping us by giving appropriate advice with the system devices and other documentation.

Abstract:

This thesis work covers the design and construction of a very contemporary advantageous electro-mechanical arrangement for fire extinguishing purpose that is an intelligent robotic system which is basically used to move through any complex maze of any building or industry where fire is swept everywhere, for the visual detection of fire in the floor of any building, for wireless transmission of data to parent server and to extinguish the fire by the manual command or run in fully autonomous mode according to the command from the base station. The proposed robot is controlled using a high power 2.4GHz ISM band radio frequency based NRF24L01+PA+LNA technology covering 1km range outdoor. The entire structure of the fire extinguishing robot was built to survive the machine in extreme heat of fire which was certainly a challenging ingredient of our research by selection of appropriate material for designing framework of the system. This multipurpose robot was used to extinguish fire which is generally occurred by indifference of inhabitants or sudden accident or unanticipated electric spark in any industry or garments as well. The proposed fire extinguisher robotic system is able to go through any difficult place by overcoming tremendous heat created by fire in any industry or fired house and efficiently transmit the video stream through setting up an automatically wireless communication between robot's microprocessor and control panels' common server. A unique RF channel is dedicated to control the robot to avoid any third party involvement into server. Therefore the proposed mechatronics for our robot, which is the sole combination of Electronics and Mechanical Engineering, can be divided in to 2 basic parts to explain the whole work easily i.e. one is Control system of a Robot by Electronic Pulse technique from a remote place and another is a Mechanically Controlling System to extinguish fire where the mechanical part consists of cylinder for fire extinguishing material like dry ice spraying mechanism. The proposed system's one of the parts could be used to control by observing wireless video transmission in a Graphical User Interface software for any other highly security purpose. To extinguish fire, high quality dry ice is used and mechanical structure is designed using computer aided drawing (CAD). For the electronics segment, a microcontroller board named Arduino is used with the corresponding firmware loaded into it. Moreover, a highly efficient fire tolerant material is used to build the body structure of whole robot in order to ensure the performance as an immaculate fire extinguisher to save humans and other valuable properties. Lastly, the experimental analysis and real life implementation of the robot is depicted all the way through work.

Chapter -1

Introduction

1.1 Introduction:

Robots are defined as electro-mechanical devices that are designed to assist people, to make critical tasks effortless but accurate execution or to use for production purpose in any industry. The foremost application of robots has so far been in the automation of mass manufacturing industries, where the identical defined tasks are to be performed repeatedly in exactly the same manner but definitely maintaining an optimum accuracy rate. However, in the recent times, there has been interest in also sending robots into such situations that are too dangerous for a person to go. For instance, buildings that are set on fire or collapsed after an earthquake needs immediate rescue operation, it is safer to send an autonomous robot into the building to investigate surrounding condition, before the rescue teams enter. There are also some other robots which are used in more dangerous circumstances like in bomb disposal, detecting mining bomb or cleaning of deadly nuclear toxic wastage.

In the event of a fire, disaster prevention is highly dependent on response time and flare control. Current smoke detector and sprinkler systems provide a fast but imprecise response. Sprinkler systems are not designed to target any specific area, so their effectiveness in fire protection is always questionable. Keeping this limitation in mind we started researching on an a remote controlled robotic system that will investigate the rough situation through video streaming by setting up a wireless communication between robot's microprocessor and control panels' common server. A problem with wireless communication is that it is not always reliable when attempting to operate a robot from outside a building. Also heavy smoke would create disturbance in indentifying the source of fire and live people. In this regards, auto-pilot mode would be activated to perform the task and make decision depending upon the information gathered from onboard electronic sensors. The device employs array of temperature sensors to localize heat sources and press the nozzle of the extinguisher or cylinder through mechanically designed switch. Ultrasound range sensor on the other hand will be utilized to discover obstacles running in autonomous mode. All of these functions are directed by arduino based microcontroller board with the help of the custom application software. As the robot is operated wirelessly from a distance by another computer or laptop with software made from Visual Studio 2010, operator can observe video, onboard sensor data in different panels. Each of the function has their own dedicated key to control from the client server i.e. activating extinguisher, moving front or backward, turning on the video.

The purpose of this research on this fire extinguishing robot is to nurture the electrical and electronics knowledge of engineering in robotics that would be a convenient application for a real-world robot in extinguishing fire efficiently and save more innocent people's lives in future.

1.2 Worldwide statistical report of fire accident:

Accident is always an unanticipated issue for any country, nation and individual. It is unlikely happened due to apathy of associated people regarding a misfortune. History bears the lethal tragedy of misfortune due to combustion. It is also unforeseen like other tragedies but the consequence is uncomfortable and demoralizing. While a lot of innocent people are trapped in any industry or residence and fire explosion starts gobbling everything around them, then it is really a dreadful moment which only the survivor could explain. If we just take a look on the history, then we will see that most of the baleful accidents and more people were died due to fire incident. December 8, 1863, The New York Times reported in their headline front page about two thousand five hundred people died because of fire accident in church [1]. That was the largest fire accident ever to have affected the city of Santiago, Chile. According to the article in New York Times, around 2,500 innocent people were burnt in Church due to a gas lamp at the top of main altar ignited some of the veils that adorned the walls. Somebody tried to put it out by something it with another cloth, but the result was not so expected. The fire jumped over other veils and from there on the roof of wood. The attendant endeavored to extinguish fire with cloak but inflammable liquid penetrated the fabric and only increased the panic. Including the above massive event, a lot of fire accidents were occurred in last decades before the eyes of people but due to lack of using advanced fire extinguishing technology, nothing was done. In the 2010, a major fire broke out in a private hospital in Hyderabad [2] where nearly 150 people were burnt to ashes. In October 29, 2012 a fire broke out through Breezy point Queens [3], NY destroying 121 homes during the Hurricane Sandy. The blaze began so high where rising Seawater contacted a house's electrical wires an Ocean Avenue and cause substantial damage to a section of town named Wedge. Below is a statistical report on casualties due to fire accident prepared on (1845 - 2012)



Figure 1.1: Worldwide statistical inspection of casualties due to fire between (1845-2012)

In fig-1.1, statistical views are given where numbers of people died are shown with their respective years. If we evaluate the data then it can be found that the losses of human life have decreased in recent years because of exercising technology in rescue operation.

1.3 History of accident due to fire in Bangladesh:

The sinister history of fire accident in Bangladesh is not also commendable. There were a lot of massive accidents due to fire occurred but due to not using modern technology people were burnt in front of associated fire fighting team. If we just take a look back on the history of fire accident in Bangladesh, then it will be clear that most of the baleful accidents were occurred in garments of Bangladesh. On the 24th November, 2012 a garment factory named "Tazreen Fashion Limited" in Nicchintapur, Savar was on fire at approximately 6PM in evening. According to the official government statistical report, the total number of death was 112 and hundreds were injured in this accident. In 2010, six people were burnt and many were injured in a fire accident in a "Mosquito coil factory" on Friday night in Dhaka, Bangladesh. On February 25 in the same year i.e 2010 fire at "Garib and Garib Limited", an export-oriented sweater factory in Savar killed minimum 21 workers and injured dozens. Including these, On February 23, 2006, a four-storied building situated at the BSCIC Industrial Area in Chittagong and used as the factory of KTS Textile and Garments caught fire and 55 workers died. Statistics also show that, 32 workers were killed at "Saraka Garments" fire accident and 20 were killed at "Jahanara fashion", Narayanganj in 1997.

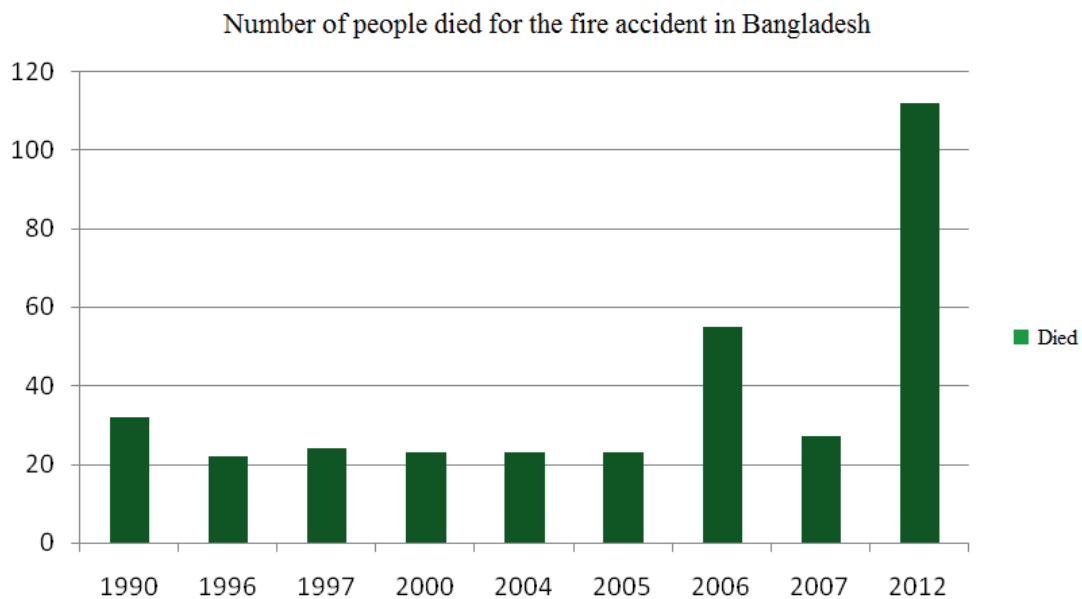


Figure 1.2: Bangladeshi statistical inspection of casualties due to fire [3]

As per this chart in fig-1.2, though the number of casualties increased than preceding years, still Bangladesh is lacking behind in using the modern fire fighting technology in rescue operation. Therefore, we settled on exploring perceptible solution to overcome from this condition by performing research on fire extinguishing automation technology which is built from materials that are available in Bangladesh.

1.4 What are Robots? :

A robot is a mechanical device, generally an electro-mechanical machine that is guided by a computer program and corresponding electronic circuitry integrated with other accessories. Each robot is built for a particular purpose. For example: A robot can be used to sort out faulty materials in any industry or robot can be used to produce cars, aero planes or computers etc. Now-a-days, the importance of robot in our life is unexplainable.

Starting from dish washing machine, stand alone CNC machine miller, unmanned aerial vehicles, humanoid robots, automated guided vehicle, wheeled robot, robotic arms, caterpillar-tracked robot etc robots are usually seen in life. However, all of the above kinds of robots are not seen in Bangladesh, most of them are used in advanced countries where their presence is now an inevitable phenomenon. Beside, fire extinguishing robot is also another category of robots whose job is to detect, send and extinguish. We named the operation of our fire extinguishing robot as DSE (Detection + Send + Extinguishing).

1.5 Importance of unmanned fire extinguisher robot:

Fire explosion never occurs spontaneously by anyone. It is a phenomenon which is occurred due to indifference, faulty or short circuit sparking of any building or industry. So, no one can predict or control this misfortune but we can use our technology to reduce the effect during spreading of fire in any building. Consequence due to fire accident is never explained because it is one of the deathliest accidents people experience in daily life. A lot of drilling of what to do when fire accident occurs anywhere can reduce the loss of lives but this could never be the ultimate solution. Every country has its own fire brigade teams who work for extinguishing purpose. This team works against fire but sometimes the situation can get worsen like fire extinguisher people couldn't enter into building or factory where fire was ignited. On that moment, expert of extinguishing fire can't assume whether any people were trapped in any floor or not. They cannot presume the inner condition of building either. To overcome from this situation, an unmanned robot can be deployed by the associated expert team to observe and put out the fire inside in any floor. By using this unmanned robot, a real time video will be transmitted from the target point to a base station where experts will be presented for monitoring the surrounding condition of ignited place. Besides, the robot will also be controlled wirelessly by the expert

team from base station. We are working for building this more efficient unmanned fire-extinguishing robot as an undergraduate thesis topic. The topic, fire extinguishing robot is selected because now a days in Bangladesh, the need for this type of robot is inevitable. Still in commercial market, this kind of robot is not solely available. Considering the deathly history of fire incident, it is now a high time and a contemporary topic for research by which we can utilize our engineering knowledge to save a lot of innocent victimized people. If our research goes in a proper way then hopefully this unmanned fire extinguishing robot will be supplied commercially in market. Though we cannot control any catastrophe or devastations but at a standstill we can use our intelligence to reduce the loss of property and lives. For reducing loss of property and lives, a fire extinguisher robot would play an important role to society. We are planning to decrease the construction cost of robot and make it as simple as possible but effective in all the sense. If construction cost is reduced then any organization regarding fire extinguishing material can buy this sort of robot and provide rescuing job in the appropriate place. Real time video transmission with manual control and full-autonomous mode are two specialties if this robot. These two modes are implemented in this robot for increasing effectiveness. Full autonomous mode i.e. auto-pilot mode will be only working in the most unfavorable condition where visual control would become impossible, that time different sensors including ultra sonic sensor, PIR sensor would be activated and start putting out fire around it.

Chapter -2

Design & Construction

Overview

2.1 Concept of Fire Extinguishing Robot

The robot which is used to extinguish fire is called fire extinguishing robot. We are researching on such fire extinguishing robot which cannot only be used to extinguish but also to monitor the ignited place and communicate with base station. While designing this robot, we kept every probable scenario in our brain that temperature outside robot would be massive like more than 900⁰C. To save our circuitry from this enormous temperature, we planned to set up a simple basic air cooling system inside the robot's chamber. Cooling system would be built using compressor of fridge and refrigerant Freon material. 2 cylinders of fire extinguishing materials are attached to the robot to douse the fire. There will be a microcontroller circuitry with wireless module and SPI communication in between them. Including motor driving circuits built by dual pole relays and MOSFETs along optocoupler for avoiding back EMF of motor and also sensor networks circuits. Sensors sent data to its base station for operator to analyze. An IP Camera is installed at the upper chamber which will transmit real time video to its particular software of base station pc/laptop. This software is developed in our research lab on the MS Visual Studio 2010 platform by using C# language for the purpose of entire monitoring and controlling the robot.

The software in our base station's computer is used from where 200 feet distant robot will be controlled manually through wireless command. To control the wheel of robot we used 2 geared dc motors and another motor is used to ON/OFF the nozzle of fire extinguishing cylinder. The mechanism is designed in such a way that after one cylinder is finished the other one would start working. Dry ice is the major component of fire extinguisher which is be sprayed towards fire to diminish it. The command for spraying is given by the expert from base station which is approximately 200 feet distant from ignited place. To monitor the surrounding condition of robot, there is a data updating section in the software installed in base station. Besides all these, sensors like ultra-sonic sensors, LM35 (Temperature Sensor), PIR sensors are attached on the body of robot. By reading the value from these sensors, any expert can identify whether any fire exists after a dense smoke or not because worse situation might arise when a dense smoke would be hindrance such a way that light or camera stops transmitting clear video stream. On that time these sensors value changing would indicate any existence of humans or fire. Fire's existence would be identified using infra-red sensor because every flame of fire bears certain percentage of IR and existence of human can also be identified by change in ultra-sonic sensors' value. Ultra-sonic receiver only receives data when the ultra sound is reflected from anywhere. If any reflection occurs from anywhere, it has a high possibility that there might be any human on danger and that time robot will automatically start spraying extinguishing materials from cylinder. Two modes are used to utilize fire extinguishing material because the material is not

abundant and its proper use is needed to be ensured. The details functionalities of the robot will be discussed on the following sections. The details of software, robot's dimension and motors' calculation are given on the following sections of this report.

2.1.1 System Architecture

The entire system can be divided into two major parts. They are:

- (a) Hardware
- (b) Software

Controlling mechanism of hardware and software are depicted below:

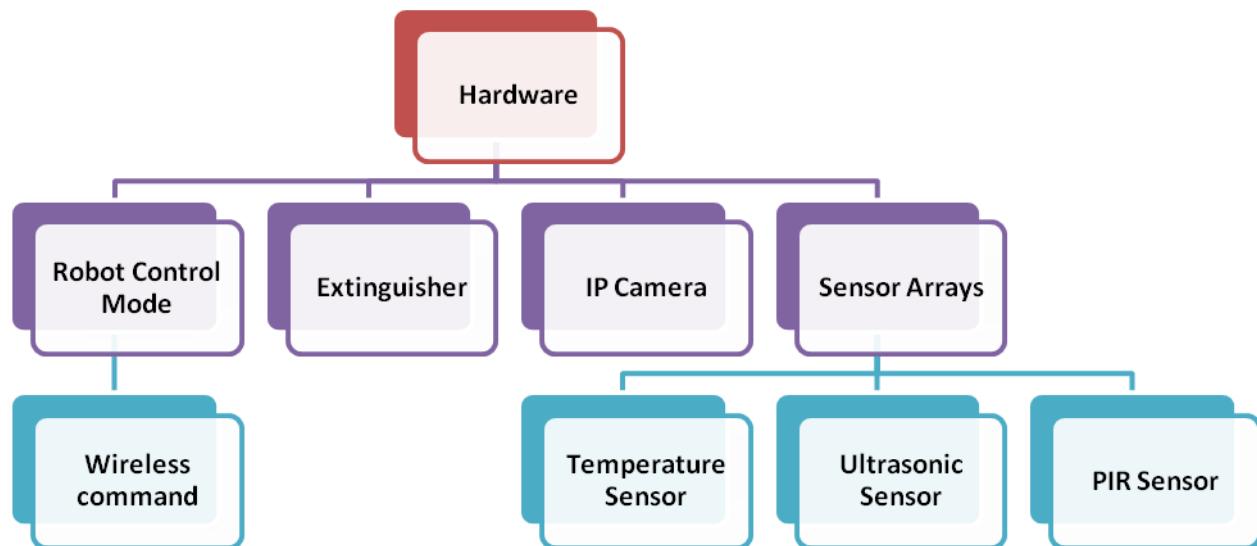


Figure 2.1: Illustration of Hardware Design

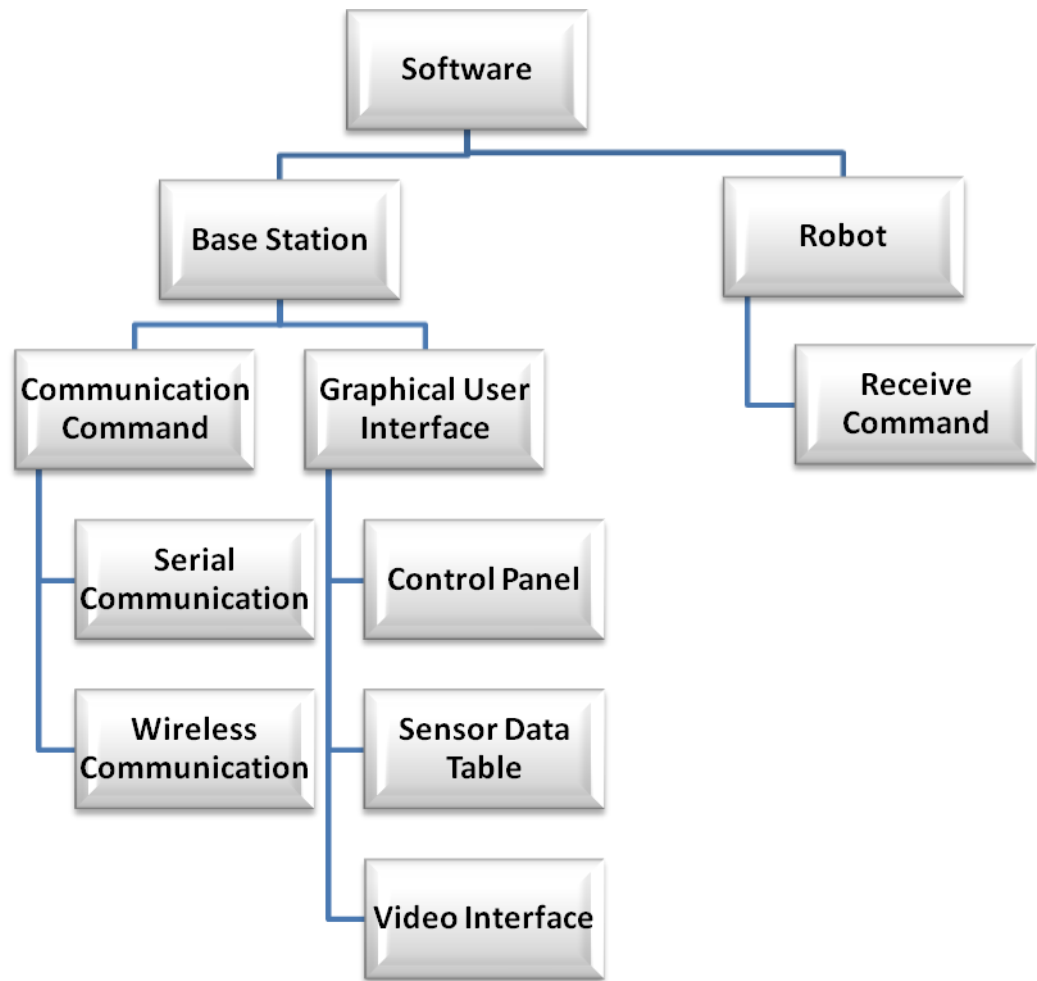


Figure 2.2: Illustration of Software Design

2.2 Mechanical Design

2.2.1 Logical aspects behind Structural Designing

We are researching on a fire extinguishing robot. This robot will be able to find out a fire in a fire affected area and extinguish it. It will monitor the ignited place and communicate with base station also. While designing this robot, we have to keep in mind that; the robot should have the ability to cross any kind of burned and narrow area, any object can fall on it, the temperature can

be very much high and the robot may have to face the fire very closely. Therefore, our designed robot should have strong chassis and its wheels should be strong enough to cross fire area and small obstacles in front of it. Whole robot body is consists of two chambers and four wheels. In the upper chamber, there are two extinguisher cylinders, sensors and an IP camera. The lower chamber covers all the control circuits, motors and power source.

The shape of the chassis can be round, triangle, square or rectangular. Among these, round shape can be selected, because it can ensure that the robot will not stuck at any corner of a bound area. A round shaped robot can move easily and smoothly around any type of bounded room. However, we have decided to include two cylinder of fire extinguishing materials inside the robot to douse the fire. Thus, a round shaped chassis will not be a good choice for us because; it needs huge space to include two cylinders. Triangle shaped chassis can provide less space-waste then round one. However, in this case the robot will have three sharp heads. At any critical area, one of these heads can be stuck hardly. Then comparing between square and rectangular shape, the rectangle is selected, because it needs less space and a rectangular chassis is enough to carry two cylinders together.

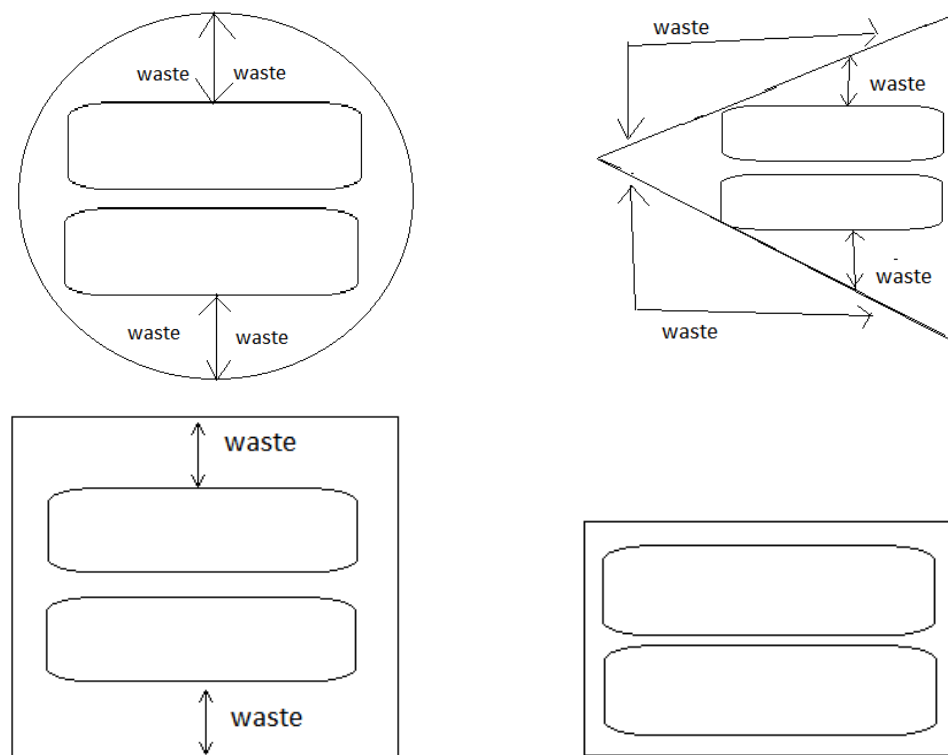


Figure 2.3: Probable Chassis Design; From Up: (Left- Round, Right- Triangle);
Down: (Left- Square, Right-Rectangular)

A rectangular chassis is the best choice because it provides highest balance of weights and motor torque, proper distribution of power and good wiring system. We divided the whole chassis into three parts; upper chamber, lower chamber and the wheel boundary.

In the upper chamber, there will be two cylinder of fire extinguishing materials, spray controller, heat, distance and obstruct sensor and an IP camera. Whole chamber is covered by a heat proofed glass and separated from the lower chamber with an aluminum sheet. The cylinders are fixed with the sheet by two “W” shaped aluminum belts. All the belts are removable, so it is possible to change cylinders after use. Two holes are made for wiring. There is also another hole for the IP camera. The cover glass is fixed with the base sheet with removable hex-screws.

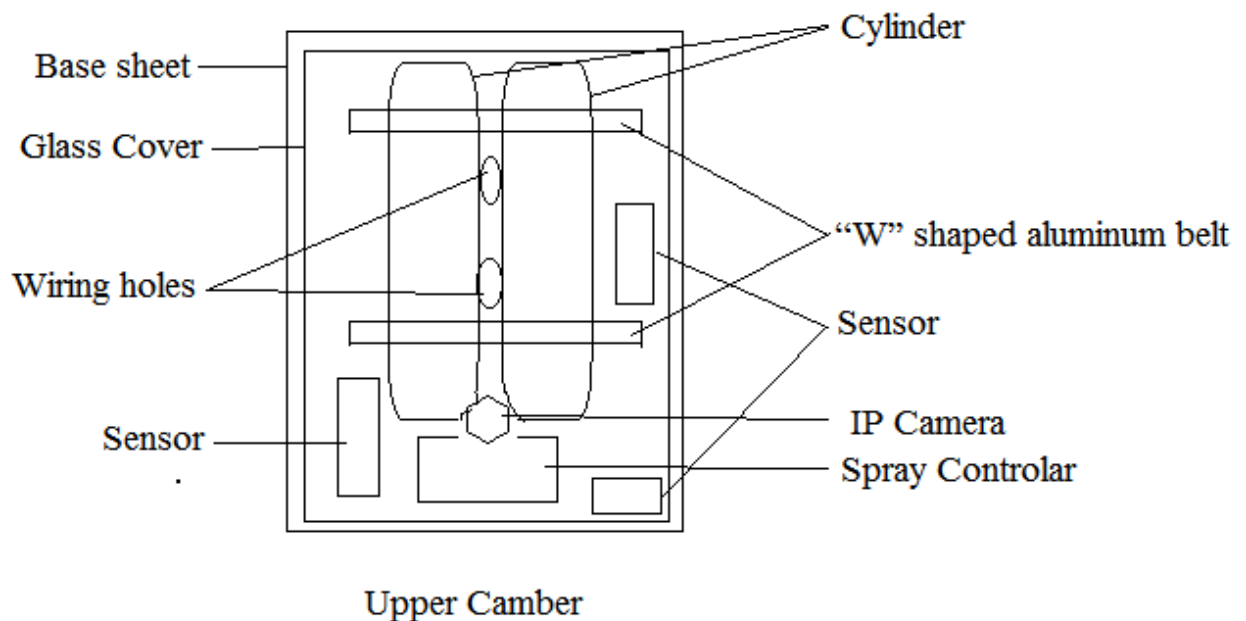


Figure 2.4: Upper Chamber of Fire Extinguisher Robot

The lower chamber will be made by aluminum sheet. While designing the robot we were cautious about the surrounding temperature. It can be very much high and can damage any circuit. Therefore we decided to make the lower chamber heat proof. Thus there will be a layer of ceramic inside the aluminum sheet. In this chamber, there are DC motors, power source, on board microcontroller based devices, and control circuits. All the devices and circuits are fixed by using removable screws. The power source is fixed by aluminum belt and the motors are

shield with the wheels. In the front side of the lower chamber, there is a slope. This slope is made for two reasons. One is to protect the camera and spray controller, and second is for having the opportunity to make any kind of change in the lower part without hampering the upper chamber.

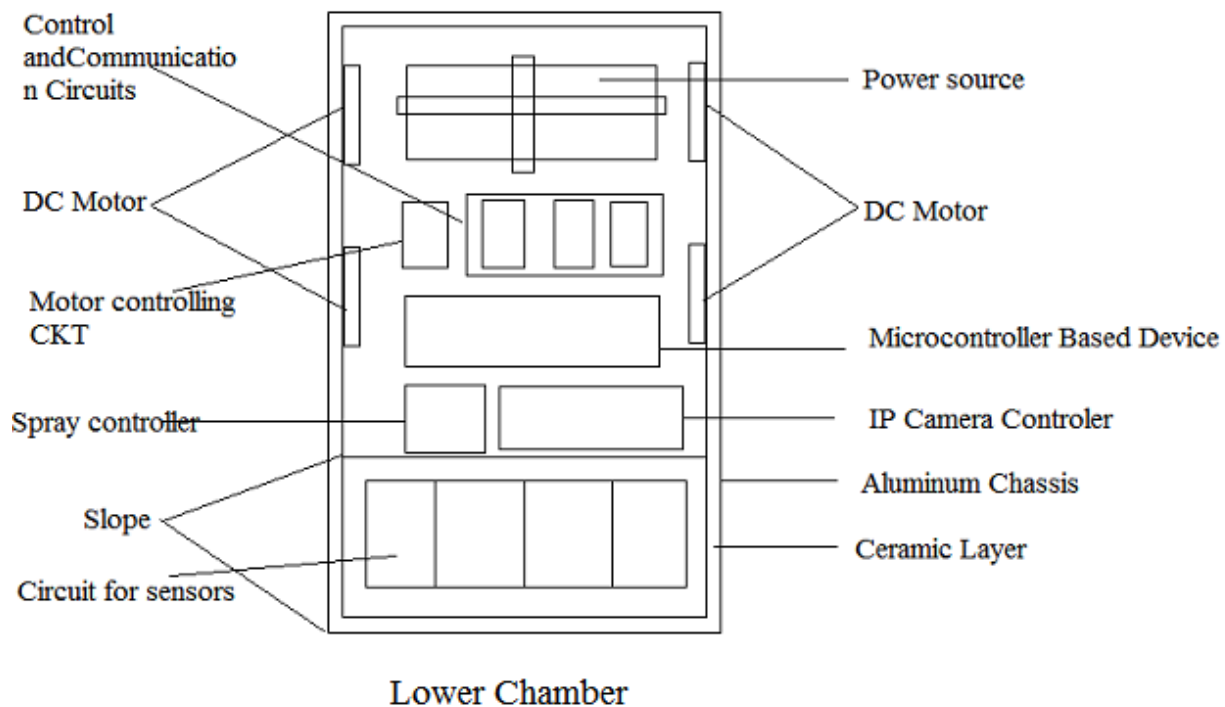


Figure 2.5: Lower Chamber of Fire Extinguisher Robot

Wheels can be set inside or outside the chassis. To make the decision about the position of wheels the robot's height, weight and length should be kept in mind. We had an approximate outline that, our robot will be 75cm in length, 45cm in weight, 35cm - 40cm in height and 50 kg in weight. We were hopeful to run it in an average acceleration of 4ms^{-2} . Thus we decided to make the wheel with a diameter of 25cm and fix it inside the chassis, but outside the chambers. The reason behind keeping the wheels outside the chamber is to keep the circuits safe and sprayed out the total mass. It ensures a smooth and fast movement of the robot.

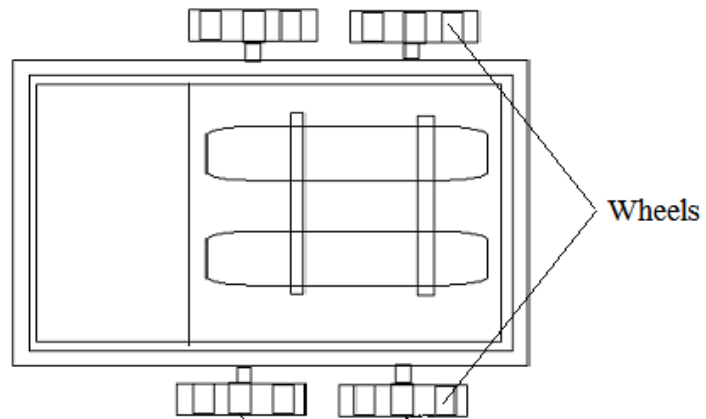


Figure 2.6: Top view of robot design

When all decisions were made about the structure of the robot, a simplified 2D model is designed to have view.

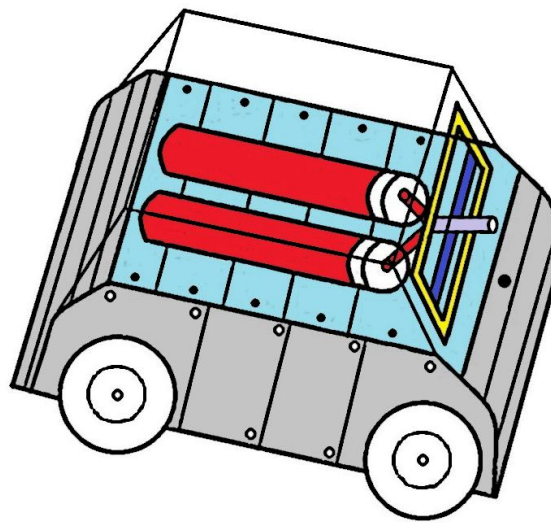


Figure 2.7: Simplified 2D model of robot with fire extinguisher

2.2.2 CAD (Computer aided drawing) Model

3D modeling is a more complete process than 2D design because of its improved quality. In conventional 2D design methods human errors such as incorrect quantities of parts would happen as the designer cannot visualize to exact scale whereas a 3D environment allows real-time simulation to visualize.

So a 3D model is made for having a clear view in all aspects. This will help us to find out, if our theoretical assumptions were right and feasible or not. Also, it gives a proper outline to the constructor. To make the three-dimensional model, we selected “Autodesk 3ds MAX”.

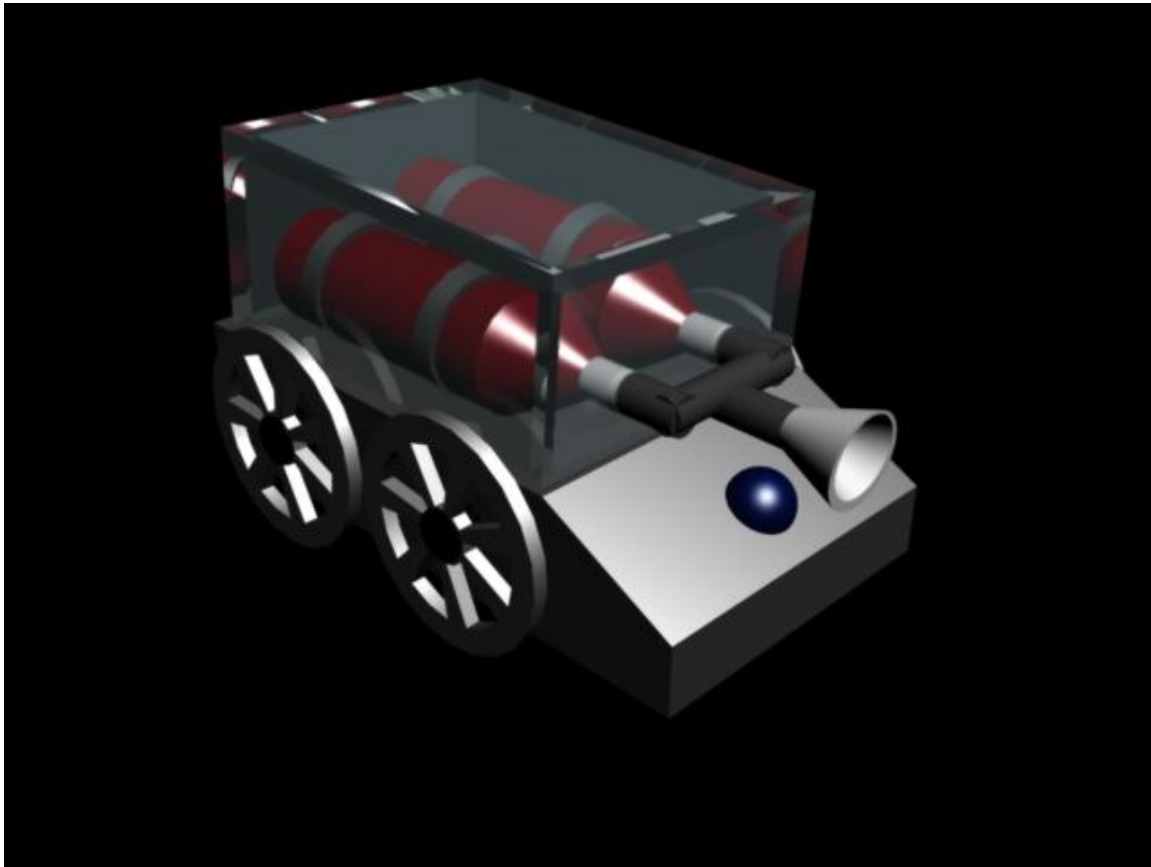


Figure 2.8: 3D model of Extinguisher robot in AutoCAD frame

2.2.3 Measurement of Robot

There are 7 basic parts of our Fire Quenching Robot. They are:

- (i) Body structure
- (ii) Wheel
- (iii) Control Circuitry
- (iv) Communication circuitry
- (v) Video transmitter
- (vi) Microcontroller
- (vii) Software
- (viii) Battery

The body structure of the robot is made of aluminum sheet and fire resistant glass. There will be a chamber in which motor, gearbox, controlling circuitry, on board laptop be fixed. We are planning to keep a compressor and simple cooling system inside robot's chamber to control heat inside the chamber. Since the surrounding temperature of robot will not be that much favorable so we must keep temperature of robot's chamber and body as low as possible. The dimension of robot is given below:

Length = 2.5'
Height = 3'
Width = 1.5'
Area = 2.5' x 1.5' = 3.75 square ft
Volume = 3.75 x 3 = 11.25 cubic ft

The estimate weight of different parts of robot is given below:

Table 2.1: Estimate weight of different parts of robot

Item	Quantity	Weight/ item	Total weight
Wheel	4	0.5 kg	2 kg
Cylinder	2	3 kg	6kg
Aluminum Body	1	10kg	10 kg
Battery	1	3kg	3kg
Glass Structure	1	2kg	2kg
Motor	2	0.5kg	1kg
Other elements	-	1kg	1kg
Total Weight			25kg

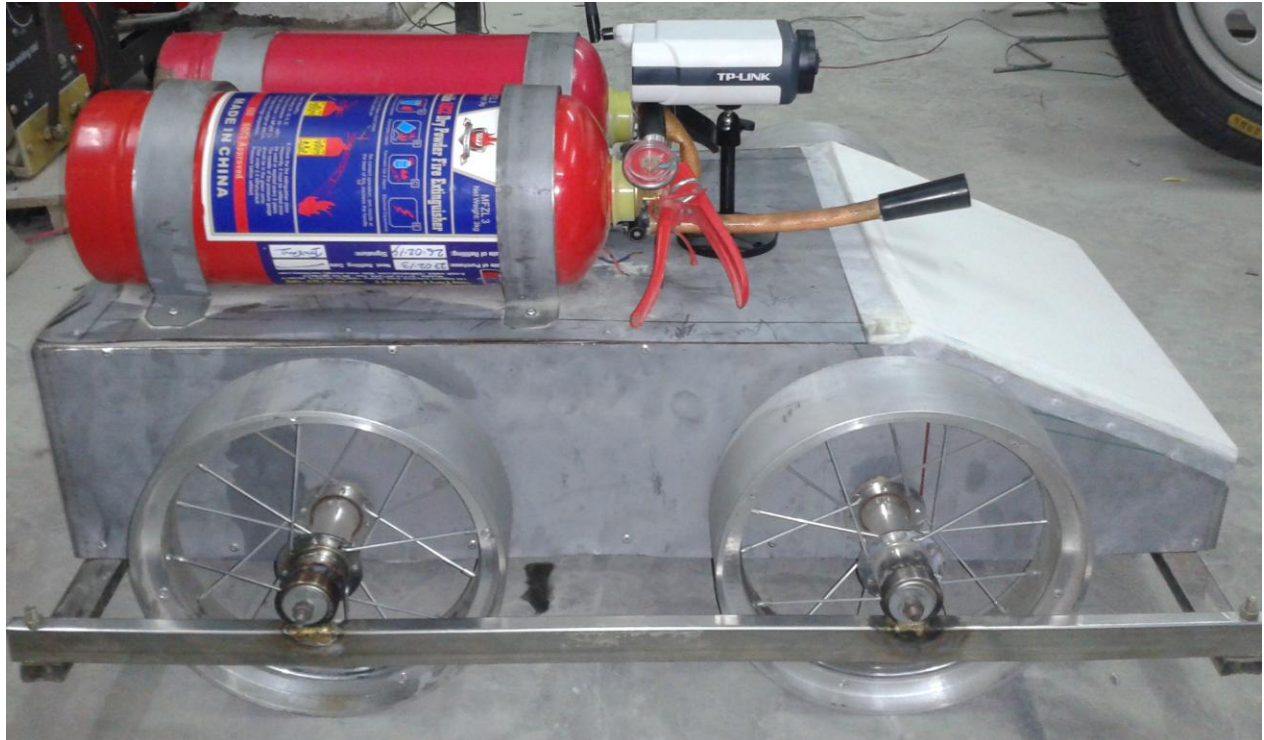


Figure 2.9: Practical Structure of Fire Extinguisher Robot

2.3 Materials for robot body

There are lots of options when it comes to take the decision to picking the building materials for robot. We know that, every material has its own characteristics, possibilities and difficulties. Different material is used to give the complete shape of the fire extinguisher robot. The main body structure of the robot is made of steel. Before selecting building materials for robot we have researched on many metals.

Metals:

1. Brass:

Brass is a substitution alloy made of copper and zinc. The proportion of copper and zinc varies upon its properties. It is quite malleable and ductile, with alloys that contain less than 35% zinc.

2. Copper:

Copper is an excellent electrical conductor. Most of its uses are based on this property or the fact that it is also a good thermal conductor. It is valued for strength, malleability, ductility, and ability to conduct electricity and heat. It is also non-magnetic, resists wear, and forms a green patina which makes it resistant to corrosion.

3. Bronze:

Bronze is an alloy consisting primarily of copper, usually with 10% tin as the main additive. It is hard and tough, and significant. Bronze parts are typically used for bearings, clips, electrical connectors and springs. It is generally more expensive than brass and more corrosion resistant.

4. Aluminum:

Aluminum is a remarkable metal with a relatively soft, lightweight, ductile metal. It has a unique combination of attractive properties. Low weight, high strength, superior malleability, easy machining, excellent corrosion resistance and good thermal and electrical conductivity are amongst aluminum's most important properties [4]. Aluminum is also very easy to recycle.

5. Steel:

Steel is an alloy of iron and a small amount of carbon [5].

Table 2.2: Characteristic comparison of different materials

Characteristic	Brass	Copper	Bronze	Aluminum	Stainless Steel
Melting Point	900-940°C	1084.62°C	2000°F	660°C	1425-1520°C
cost effective	Quite expensive	Quite expensive	Expensive than Brass	Cheap	Very Cheap
Strength to Weight	Better than copper	Low		High	High
fire resistance capability	Low	High	Medium		High
electrical and thermal conductivity	23-43% of the conductivity of Copper which is High.	Highest Conductivity		High	Poor

Weight				One third of steel, 2700 kg/m ³	
--------	--	--	--	---	--

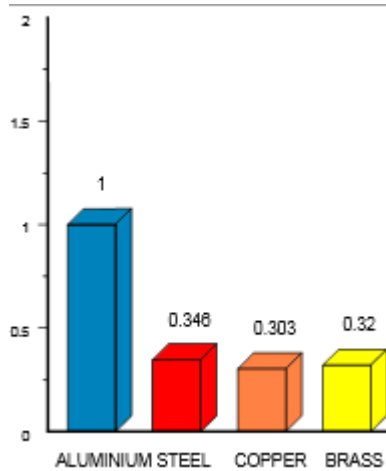


Figure 2.10: Volume per unit weight of Aluminum, steel, copper, Brass [4]

Wires:

We have used lot of wires which are made of copper. Copper is very good conductor. It has been used for as a wire in connection of different circuit points. Benefits of Copper wiring:

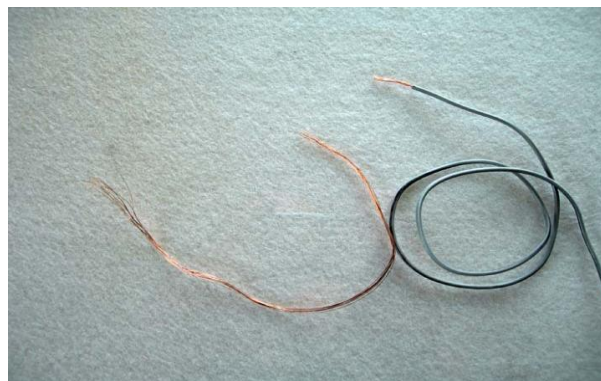


Figure 2.11: Copper Wire

1. Best Conductor: Out of all the different metals and wire out there, copper is the best conductor out of all of them.
2. Flexibility: When we are using an electrical wire, there is a good chance that we need to be flexible. We most likely have to need to bend the wire around several different things.

3. High Melting Point: It takes a lot to melt a copper wire.

Bearings:

We have used four bearings in wheels base in fire extinguisher robot and these bearing are made of bronze. The logic behind this is bronze is heavy. Bronze is more corrosion resistant.



Figure 2.12: Bronze bearing

Wheels:

Four wheels of robot are made of aluminum. Since our robot is medium sized it is better to use aluminum. Aluminum is commonly available in extrude forms in different shapes. At temperature of 200 Degree Fahrenheit its thermal conductivity is 124 Btu/ (hr °F ft).

- In addition, there are many reasons for selecting aluminum like-
- Light weight: Aluminum is a very light metal with a specific weight of 2.7 g/cm^3 , about a third that of steel. It reduces dead-weight and energy consumption while increasing load capacity.
- Corrosion Resistance: Aluminum naturally generates a protective oxide coating and is highly corrosion resistant.
- Electrical and Thermal Conductivity: Aluminum is an excellent heat and electricity conductor and in relation to its weight is almost twice as good a conductor as copper.
- Strong
- Easy to work with
- Melting point is 666 Degree Celsius

- It is very pretty cheap

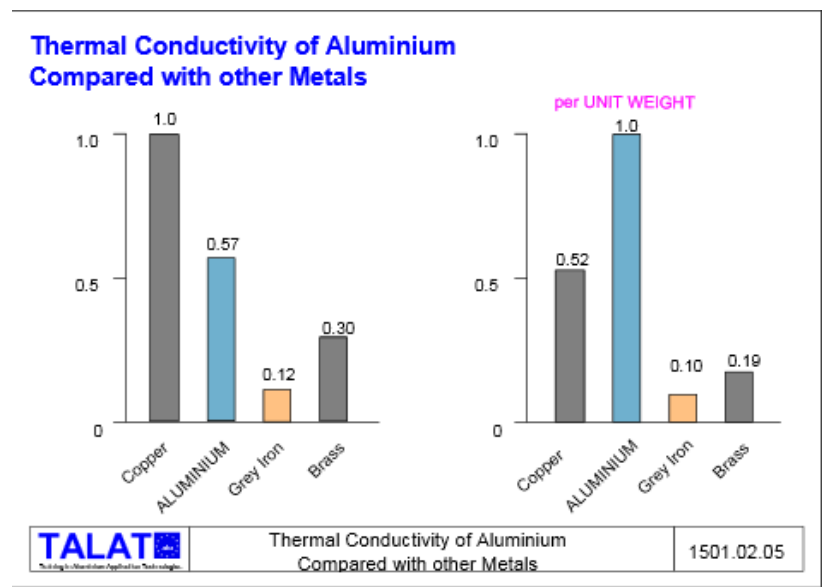


Figure 2.13: Thermal conductivity of Aluminum Compared with other Metals [7]



Figure 2.14: Aluminum Wheel base

Frames:

For making different sizes of frames we have used steel as a major materials of robot body. Commonly available steel is an alloy of iron. We have used steel because there are many features of steel for example:

Cost Effective and quick to build: with FRAMECAD'S steel frame components, construction is quick and simple allowing low-cost.

Strong and Design- Flexible: Steel offers architectural and design flexibility due to its inherent strength. This allows large span distances and curves to be easily incorporated into designs.

Fire Resistant: Melting 1425-1540 degree Celsius [5]

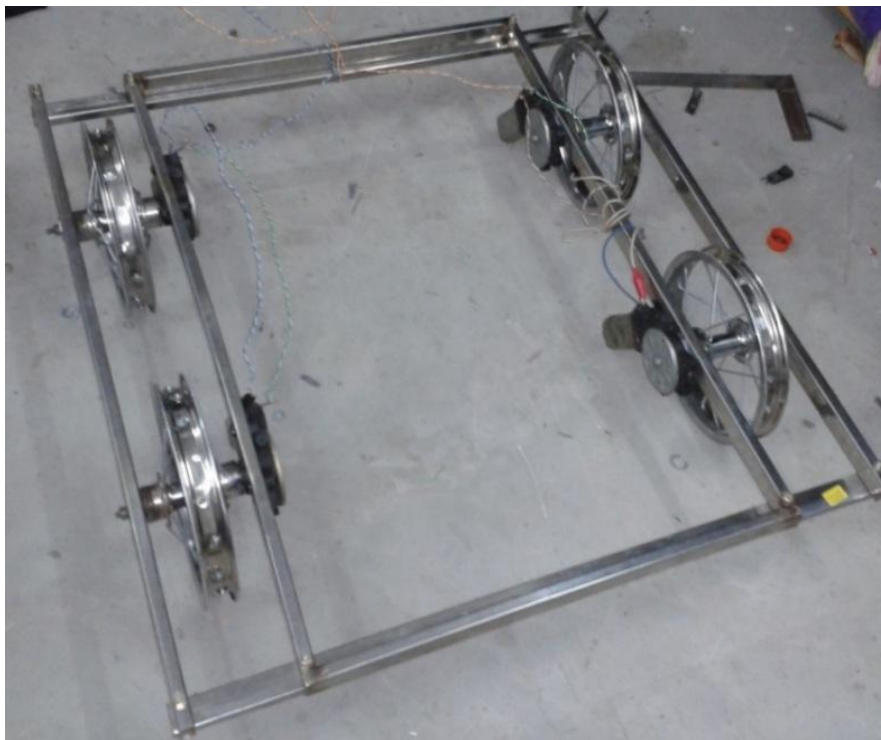


Figure 2.15: steel frame for robot

Fire Resistant Glass:

We have made a prototype for roof of the robot body since there will be temperature on firing place. While selecting suitable glass as cover we gave importance of less heat transmission and after analyzing the table, it is better to use fire resistant glass as roof.

Table 2.3: Effects for Radiant Heat

Glass Type	Radiant Heat (KW/ SQ. meter)
Fire resistive	<1
Heat reflective	1
Wired glass	39-48
Ceramics	75

Fire resistant glass is of two types [6]:

1. Heat transmitting glass: Able to prevent flame and inflammable gas to pass however unable to stop heat transmission over the glass.
2. Fire-insulating glass: Contains flames and inflammable gas for a longer period of time and prevents the transmission of flames and smoke as well as heat.

It has below features:

1. Multi-laminated fully insulating fire-resistant safety glass
2. Offers the highest level of fire protection
3. Effectively blocking transmission of conductive and radiant heat
4. Maximizing the transmission of natural and transparency [6]

Since Insulated glass do not allows radiant heat transmission, the glass should be used in this project to avoid the enormous heat.

2.4 Primary Equipments

2.4.1 Microcontroller Board:

Arduino is an open-source electronic prototyping platform for building electronics projects. Arduino consists of both a physical programmable circuit board designed around 8-bit Atmel AVR microcontroller and a piece of software, or IDE (Integrated Development Environment) including a programming language compiler and a boot loader that runs in computer, used to write and upload computer code to the physical board [8].

There are many types of arduino-compatible and arduino-derived boards. Some are functionally equivalent whereas some are interchangeable. Till date sixteen versions of Arduino hardware have been commercially produced and some of them are: Arduino uno, Arduino Mega, Arduino ADK, Arduino Leonardo, Arduino nano, Arduino Due, Lilypad Arduino.

On the otherhand, Arduino IDE is a cross-platform application written in java and is capable of compiling and uploading programs to the board with a single click. Arduino programs are written in C or C++ language.

Why Arduino? :

There are many other microcontrollers and microcontroller platforms available for physical computing- Parallax Basic Stamp, PIC Microcontroller, ATmega, Netmedia's BX-24, MIT's Handyboard, and many others offer similar functionality. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware called programmer in order to load new code onto the board as a USB cable can do that task. Because of its flexibility and various functional pins also added up to the mark.

Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. It is published as an open source tool that can be available for extensions Finally Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

Moreover to that, Arduino IDE supports windows, Linux, Mac operating systems.

Arduino used in This Project:

Since our project consists of wireless control system, two microcontrollers-one for robot and one for operating area, are needed to accomplish the whole development. Thus Arduino Mega2560 is used in operator's electronic hardware, where on the other hand Arduino ADK is employed in onboard robot.

2.4.4.1 Introduction to Arduino ADK:

Arduino ADK is Atmega2560 based microcontroller board where there is a USB host for Android phones. It also has a USB-to-serial converter. To protect short circuit and over currents it has a resettable polyfuse that provides extra layer of protection. If more than 500 mA current is applied fuse will break the connection till the overload removed [9].

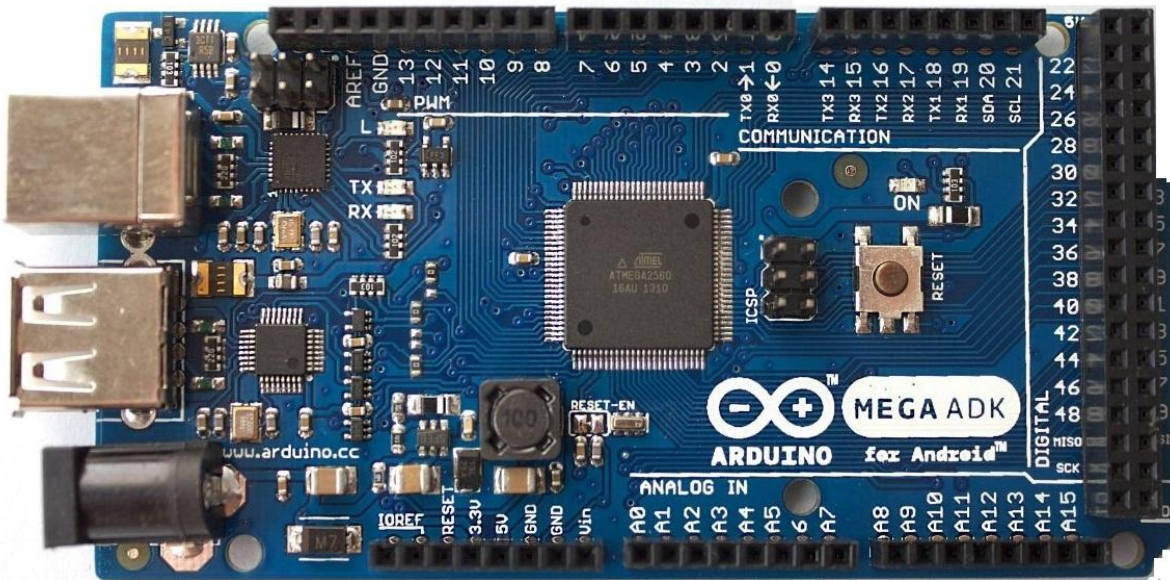


Figure 2.16: Arduino ADK board

Features:

- Microcontroller: ATmega2560
- Operating Voltage: 5V
- Input Voltage : 7V-12V
- Digital I/O pins : 54
- Analog I/O pins : 16
- PWM Pins : 15
- SRAM : 8 KB
- EEPROM : 4KB
- Clock Speed : 16MHZ
- DC Current per I/O pin : 40mA
- DC Current for 3.3V pin : 50mA

Functions of Pins:

- Serial Communication pins: Serial 0 (RX-0, TX-1), Serial 1(RX-19, TX-20), Serial 2 (RX-17, TX-16), Serial 3 (RX-15, TX-14)
- External Interrupts Pin No.: 2,3,18,19,20,21. In total 6 pins for interrupts.
- SPI : 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS)
- TWI : 20 (SDA), 21 (SCL)

Special Functions Pins Used in this research:

Because of multifunctional activities of pins, Arduino lessens some burden of using multiplexing, de-multiplexing, decoding-encoding techniques. In this research work Arduino ADK is used as an operating hardware, most significantly SPI pins are used for connecting with wireless module named NRF24L01.

SPI means serial peripheral interface that operates in full duplex mode. It is a synchronous serial data protocol used by microcontroller for communicating with other peripheral devices. In a SPI connection there is always one master device that controls peripherals devices and thus there are three lines common to all devices.

MISO (Pin 50): MISO means master input slave output.

MOSI (Pin 51): MOSI means master output slave input

SCK (Pin 52): Serial clock

SS (Pin 53): Slave select

SPI-a master slave protocol needs only four pins from Arduino to communicate. A master device control data communication where data is controlled with a clock signal [10]. The slave select pin (SS) control is used when there is more than one slave present.

2.4.4.2 Introduction to Arduino Mega 2560:

Arduino Mega 2560 is based on ATmega2560- an 8 bit Atmel microcontroller with 256K byte system programmable flash. It has 54 digital I/O, 16 Analog I/O, 4 UARTs, ICSP header and a reset button. From all preceding boards the Mega2560 did not use the FTDI USB-to-serial chip rather it features the ATmega16U2 programmed as a USB-to-serial converter [11].

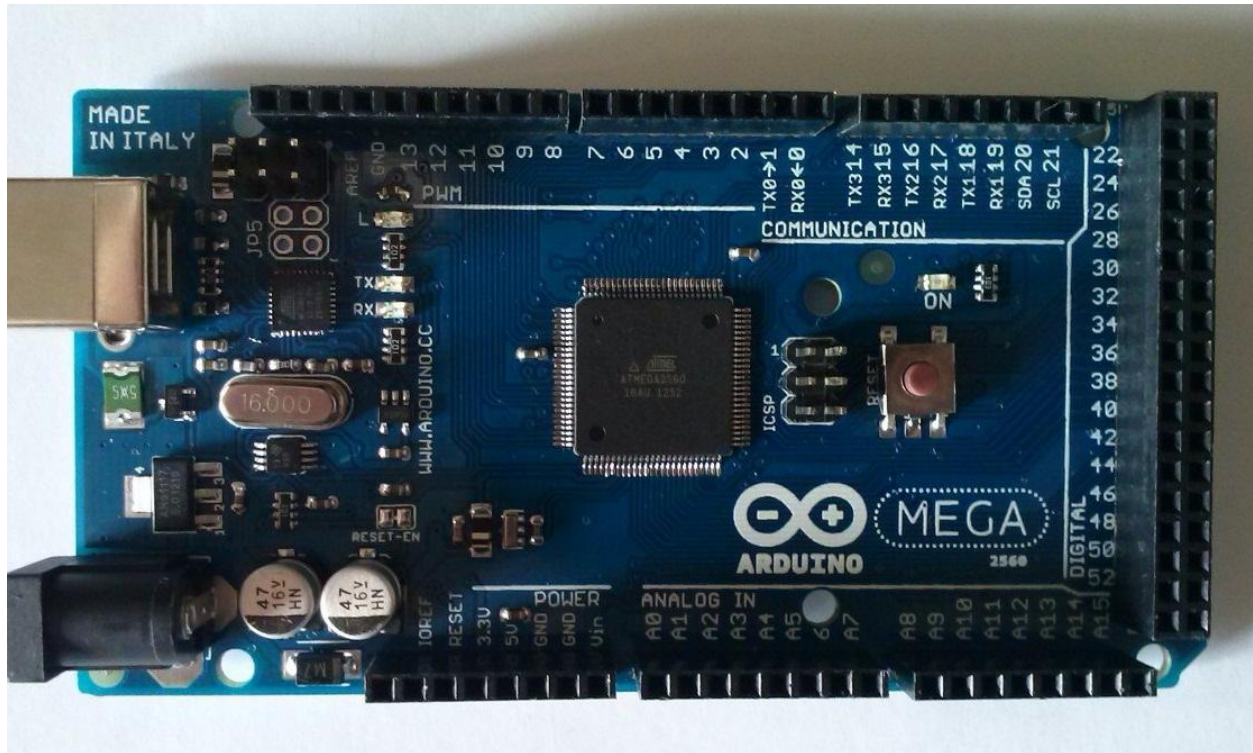


Figure 2.17: Arduino Mega 2560

Features:

- Microcontroller: ATmega2560
- Operating Voltage: 5V
- Input Voltage : 7V-12V
- Digital I/O pins : 54
- Analog I/O pins : 16
- PWM Pins : 15
- SRAM : 8 KB
- EEPROM : 4KB
- Clock Speed : 16MHZ
- DC Current per I/O pin : 40mA
- DC Current for 3.3V pin : 50mA

Special Functions Pins Used in this research:

Arduino Mega2560 microcontroller board is used in onboard robot and this hardware is going to operate several sensors, a servo motor and 4 relay controlled motor circuits. So, selecting constructive microcontroller based hardware was necessary.

PWM Pins:

In Arduino Mega2560 15 PWM pins are available and those are Pin No. 2-13. PWM indicates for pulse width modulation that uses in controlling motor speed. PWM techniques refer to controlling supply voltage to motor controlling circuit directing according to command. Also one PWM pin is used in sensing ultrasonic sound to detect obstacle and identifying its distance from robot. In that case PWM pin send a pulse and then listen eco pulses to calculate further.

SPI:

NRF24L01 wireless module is connected with the 4 pins of SPI. The wireless module NRF24L01 that is used in this research work is a transceiver thus it has to read data at the same time send data to the microprocessor based Arduino Mega2560. Since the wireless module uses serial peripheral interface, a master-slave protocol is followed, thus uses these 4 pins. Also, the wireless module uses extra 3 pins in arduino.

Digital I/O pins:

4 digital I/O pins are employed to control electrical devices. One of them is PIR sensor as it has to send a signal to let the Arduino know the changes of thermal activity of surroundings. A servo motor uses digital I/O to act according to remote command, on the other hand to activate solenoid valve for fire extinguishing corresponding relay has to be activated, therefore uses a digital I/O pin.

Analog I/O pin:

For knowing the surrounding temperature analog pin is used to get the reading from LM35. Since LM35 give analog value ranging from 0-1023, a pin from ADC port of Arduino Mega2560 is used.

2.4.2 Optocoupler

In electronics, a device that uses optical transmission path to transfer signal while keeping them electrically isolated is named optocoupler. Optocoupler technology is used in variety of isolation applications i.e. power supply, motor control circuits, digital interface circuits. Optocoupler eliminates electrical noises, interference being a cost effective device.

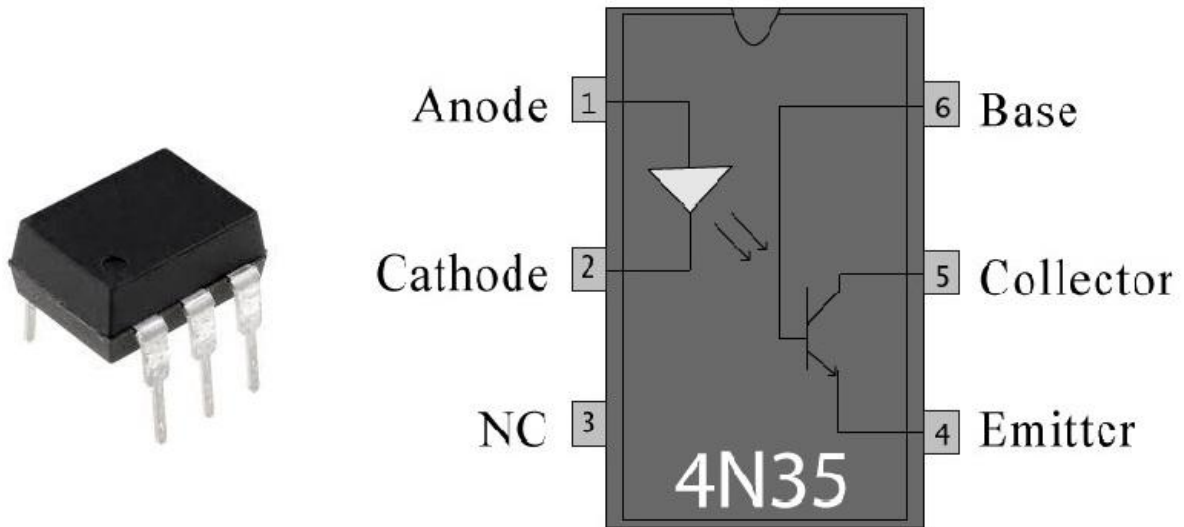


Figure 2.18: Internal Structure of Optocoupler (4N35)

Features [12]:

- High Current transfer ratio
- High isolation voltage

2.4.2.1 Working Principle:

Optocoupler is a component connector that works on optical light trigger. Optocoupler consists of transmitter and receiver parts. The transmitter is built from an infrared LED where the receiver is built with the basic components of Photodiode. Photodiode is one kind of power transistor sensitive to light source. A light source produces heat energy, as well as the infrared spectrum. Thus photodiode is more susceptible to capture radiation of infrared rays.

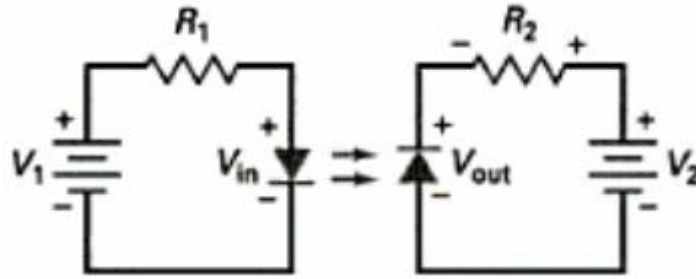


Figure 2.19: Working principle of Optocoupler

When an electrical signal is sent to the input, LED lights and causes generation of electrical signal at the output. Because of the photodiode the output current is proportional to the amount of incident light supplied by the emitter [3]. This set up a reverse current in output circuit that generates voltage across resistor. However the output voltage then equals to the supply voltage minus the voltage across resistor [13]. That means,

$$V_{out} = V_2 - V_R$$

2.4.2.2 Purpose of 4N35 in This Project:

As one of the most useful purposes of optocoupler is to provide protection from high voltage and low level noise, it also defends current back to its input stage.

In this study, we require a device that can protect the microcontroller from the back EMF of the motors. Since in optocoupler, the transmitter and receiver circuit is 100% isolated from each other, it saves the circuit from static electricity, relay and motor back-EMF damage [14].

2.4.2.3 Experimental Analysis:

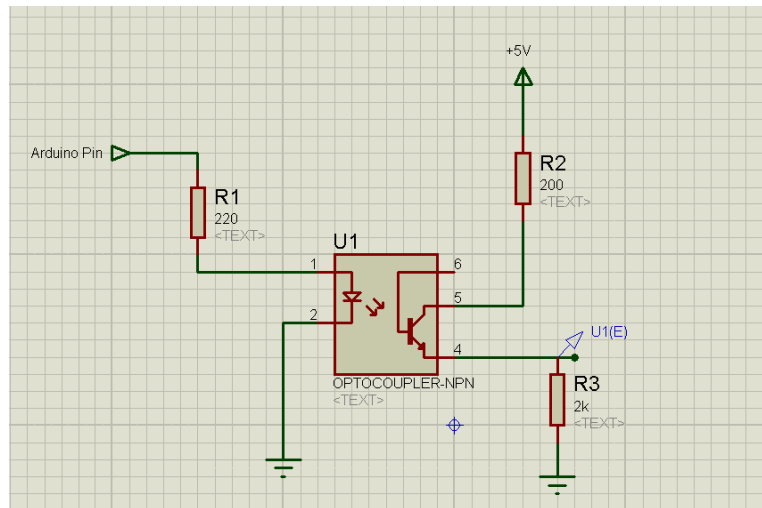


Figure 2.20: Connection diagram of 4N35 with Arduino Mega in ISIS 7 Professional

Since we need to control motor controlling circuit through arduino optocoupler is used to isolate the Arduino from the controlling circuitry. According to figure, pin1 of 4n35 is connected with arduino for getting signal to pass to the other side. Whenever the anode pin get any voltages the light source produces heat energy and pass over the same amount of voltage to the emitter pin. In this project the range of voltage to send is between 0-5V and so collector pin is connected with a power source of 5V.

2.4.3 Power MOSFET

Power MOSFETs (Metal-Oxide Semiconductor Field Effect Transistors) are three-terminal silicon devices that function by applying a signal to the gate that controls current conduction between source and drain. They are most commonly used power devices due to their low gate drive power, fast switching speed and superior paralleling capability. Most power MOSFETs feature a vertical structure with Source and Drain on opposite sides of the wafer in order to support higher current and voltage. In this thesis research power MOSFETs are needed in controlling actuators like motors, solenoid valve. These actuators need high current and voltage to operate whereas arduino will feed only 5V. So, actuator controlling circuits are designed in such a way that these power MOSFETs will switch on for a low gate voltage to activate the relays and for this n-channel IRFP250N MOSFETs are chosen to use.

2.4.3.1 IRFP250N:



Figure 2.21: IRFP250N MOSFET

Features[15]:

- 175°C operating temperature
- Fast switching
- Fully avalanche rated
- simple drive requirements
- Dynamic dv/dt rating

2.4.3.2 IRFP250N characteristics:

Breakdown Voltage:

Breakdown voltage, BV_{DSS} , is the voltage at which the reverse-biased body-drift diode breaks down and significant current starts to flow between the source and drain by the avalanche multiplication process, while the gate and source are shorted together [16]. According to the datasheet of IRFP250N breakdown voltage is 200V for 250 μ A drain current.

On-Resistance:

On-Resistance ($R_{DS(ON)}$) of the power MOSFET determines the power loss and heating and the lower the device power loss and the cooler it will operate. Low on-resistance drastically reduces heat-sinking requirements which lower parts count and assembly costs. In IRFP250N power

MOSFET device the static drain-to-source on-resistance is 0.075Ω on conditions of gate-to-source voltage 10V and drain current 18A.

Threshold Voltage:

Threshold voltage $V_{GS(TH)}$ is defined as the minimum gate bias which can form a conducting channel between the source and drain [17]. For power MOSFETs, it is usually measured at the drain-source current of 250uA. Gate oxide thickness and doping concentration of the channel can be used to control the V. Under the condition of $V_{DS} = V_{GS}$ and $I_D = 250\mu A$, gate threshold voltage for IRFP250N is 2-4V.

Power Dissipation:

Maximum Allowable Power Dissipation, P_D , describes the MOSFET maximum dissipation for the maximum allowable junction temperature, $T_{J(max)}$. Typically, this rating specifies the case temperature held at 25°C. $T_{J(max)}$ is normally 150°C or 175°C.

$$P_D = \frac{T_{jmax} - 25}{R_{thJC}} [16]$$

In IRFP250N datasheet power dissipation for 25°C is 214W.

2.5 Relay:

The relay is an electrically controlled device that opens and closes electric contacts affecting other devices in the same or another circuit. When the switch is closed, north and south pole are produced across the working gap by solenoid winding of the coil, producing a magnetic flux. Note that power source shown is direct current. If the current in the coil is slowly increased from zero, a point will be reached where the armature snaps (closing the gap) and closes the normally open (N.O) contact. This is defined as the pick-up point and is specified by current or voltage by the manufacturer. Conversely, if the current in the coil is gradually decreased, a point where the flux is too weak to hold the gap closed is reached and the armature snaps open, this is called the *drop-out current* and is usually considerably less than the pickup current or voltage. The only noticeable physical difference between AC and DC relays with the same rating is the addition of a shaded pole at the end of the core. A shaded pole is a conducting ring that creates a magnetic field that lags the applied field, prohibiting the rapid collapse of the flux across the gap. If this were not the case, the relay would hum or chatter at a 60 Hz rate, and contact positioning would be indeterminate.

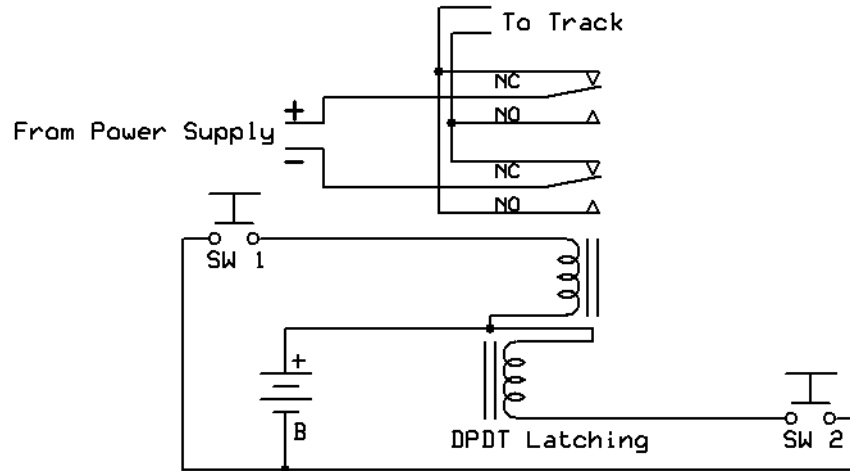


Figure 2.22: Basic circuit diagram for driving a dual pole relay

Contact arcing is more common in dc than in ac circuit interruptions. The ac circuits go through zero voltage at each half cycle and extinguish any arcing that occurs. Any metal transfer is generally eliminated, except for a roughening of the contact faces. The dc arcing or spark discharge is damaging and will cause metal to transfer from the negative contact to the positive contact, as shown in Fig 4.14. The needle-type transfer which occurs at low voltage and low contact force is particularly objectionable, and the needlelike buildup may cause interlocking of the contacts. To prevent this, contact shunting or arc suppression may be used to eliminate or minimize contact arcing when switching inductive loads. When the contact opens, the inductive load generates a voltage due to its collapsing field. This voltage will cause the capacitor to charge, and arcing will be avoided. The following formulas are applicable for determining R (ohms) and C (microfarads) for the RC suppression network across the contacts:

$$C = \frac{I^2}{10}$$

where,

I = circuit current, A

C= capacitance, μF

$$R = \frac{0.1 V}{I^x}$$

Where, V= Open circuit voltage (induced + supply)

$$X = 1 + \frac{50}{V}$$

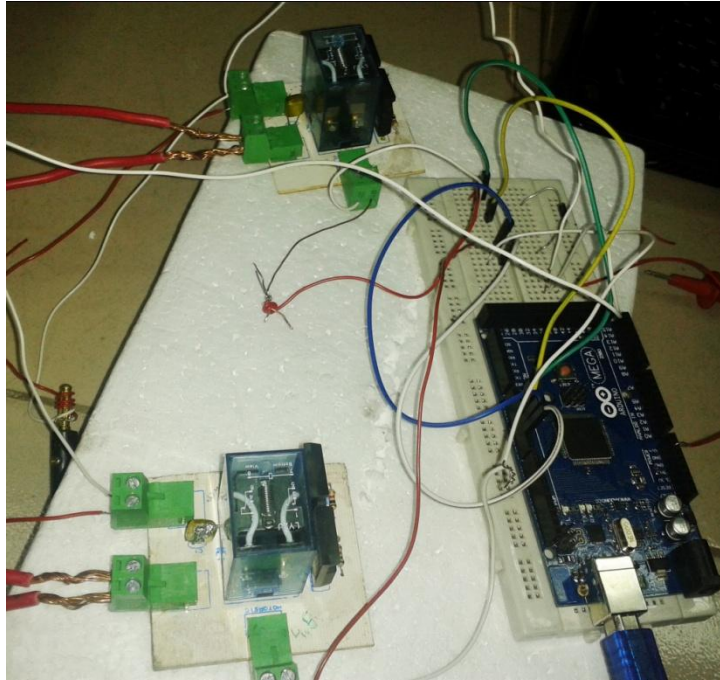


Figure 2.23: Experimenting with Dual pole Relay and Arduino Mega

The above demonstration shows experiment with relays for FireBOT. While experimenting we faced problem with RF interference. To nullify interference, the below procedure was maintained.

Arcing also causes radio frequency interference (RFI) signals. Because RFI signals can cause problems with the operation of sensitive equipment and instruments, these suppression devices are very important for the role they play in reducing interference. The most effective point of installation is at the source or at the contacts.

Another problem with relays relates to the collapsing magnetic field of relay coil , generating a transient or cemf voltage expressed as

$$V = \frac{Ldi}{dt} v$$

where, di is the change in current

dt is change in time

L is the inductance in Hentries (H)

Chapter -3

Study of Motors for FireBOT

3.1 Motor

We decided to make the robot as simplest way as possible, but tried to ensure the lowest power consumption, lowest torque loss, highest speed and smoothest rotation. Thus we decided to use “Geared DC motors” to run the wheels and a “Servo motor” to rotate the IP Camera.

It was possible to use only one DC motor to run the wheels, but it may arise several problems during any type of rotation, climbing slop or passing over any obstacle etc. These problems can be solved by using two separate motors for two sides. However, there are some critical problems in balancing the wheels of one side with only one motor. Therefore we decided to use four separate motors for four wheels. Use of four separate motors ensures the smooth and speedy movement of the robot. Only special thing we have to use is a differentiator.

Then the question arises is, which kind of DC motor should be used? We made some survey to have a clear idea about the atmosphere of fire affected areas. Some of the surveys are, December 8, 1863, The New York Times reported in their headline front page about two thousand five hundred people died because of fire accident in church [18]. In the 2010, a major fire broke out in a private hospital in Hyderabad, where nearly 150 people were burnt to ashes [19]. In October 29, 2012 a fire broke out through Breezy point Queens [20], NY destroying 121 homes during the Hurricane Sandy. From the surveys, we found that, if we send our robot into any fired building or room, it may have to make a hard break to safe itself. Thus we decided to use “Geared DC motor”. Gear reduction produces the torque needed at a steady, but slow, speed [21]. Using geared motor, we can provide a good speed at any type of surface (plane or slope). It also ensures that, the robot can off its movement in a certain; even the robot can be sucked on a slope.

While selecting motor for a robot, weight is another point of concern. Some may look for light motors. Standard DC motor is an option that can be used but our main motive is to make the robot as light as possible [22]. However, we gave it a small priority. Because, we are using four motors and have no tension about the weight of the robot.

A tradeoff between speed and easy control system may occur. DC motors make more vibrations then servo motors at the time of collisions, servo motors are small in size and easy to control. Since we decided to make our robot rectangular in size, we have enough space to keep the motors inside the chassis. Thus the big size of DC motors is not a problem for us. Then the way of control comes. Though it is difficult to control DC motors, we made a good design and made it easy. We considered the difficulties only due to have enough torque.

At the very beginning, our assumed torque of the motor was 57Nm. At that time we thought about a robot of 25kg. 58.83 W power is needed to provide this torque. Our targeted acceleration of the robot is 4ms^{-2} . To provide this, the motor needs to have 11 Rotation per Minute and angular velocity of 1.11 red/sec.

After some modification, we found the weight is 45kg and the torque is 92Nm. 138W power is needed to provide this torque. Also motor needs to have 15 Rotation per Minute and angular velocity of 14 red/sec.

Before selecting the type of motor, we had to find out, if it is possible to provide such huge amount of power to the motors. We made some calculation and found; only DC motors can work safely in this huge power and provide this huge torque.

3.1.1 Motor torque and RPM calculation

Torque Calculation:

Consider,

Mass of Robot, $M = 25\text{ kg}$

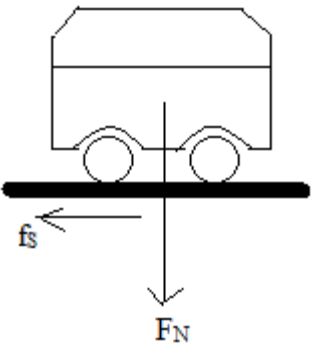
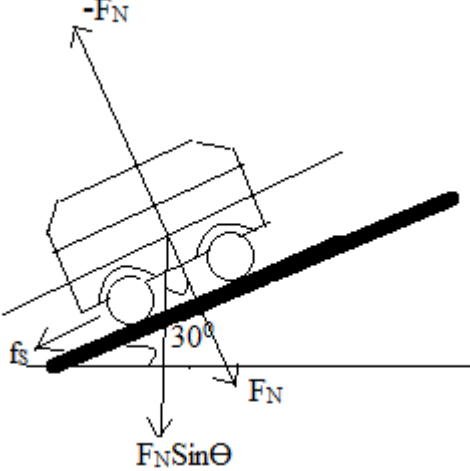
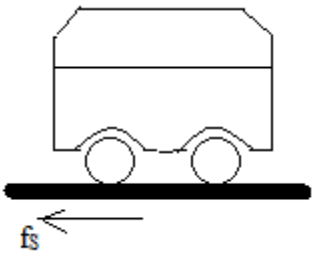
Frictional Coefficient, $\mu = 0.80$

Acceleration, $a = 4\text{ ms}^{-2}$

Gravity, $g = 9.81\text{ ms}^{-2}$

Coefficient of Rolling Resistance, $C_{rr} = 0.303\text{ ms}^{-2}$

Radius of wheel, $r = 0.127\text{ m}$

Starting with acceleration a in a horizontal floor	Climbing 30 ⁰ Slope with acceleration, a	In a horizontal floor in a constant velocity (a=0)
 <p>Fig.3.1</p>	 <p>Fig.3.2</p>	 <p>Fig.3.3</p>

Torque, $\tau = Fr$

Force calculation:

Stage (1): Starting with acceleration **a** in a horizontal floor (fig.3.1)

$$\begin{aligned}F(1) &= F + f_s \\&= Ma + \mu Mg \\&= (25 \times 4) + (0.80 \times 25 \times 9.81) \\&= 296.2 \text{ N}\end{aligned}$$

Stage (2): Climbing 30° Slope with acceleration, **a** (fig.3.2)

$$\begin{aligned}F(2) &= F + f_s + F_N \sin \Theta \\&= Ma + \mu Mg + Mg \sin \Theta \\&= (25 \times 4) + (0.80 \times 25 \times 9.81) + (25 \times 9.81 \times 0.5) \\&= 418.825 \text{ N}\end{aligned}$$

Stage (3): In a horizontal floor in a constant velocity ($a=0$) (fig.3.3)

$$\begin{aligned}F(3) &= f_s \\&= \mu Mg \\&= (0.80 \times 25 \times 9.81) \\&= 196.2 \text{ N}\end{aligned}$$

Table 3.1: Calculation of torque for different force

Force	Torque, $\tau = Fr$
F (1) = 296.2 N	$\tau (1) = 39.8 \text{ Nm}$
F (2) = 418.825 N	$\tau (2) = 56.4 \text{ Nm}$
F (3) = 196.2 N	$\tau (3) = 27.1 \text{ Nm}$

Observe that, $\tau (3) < \tau (1) < \tau (2)$

Let consider the maximum value of torque.

$$\tau = \tau (2) = 56.4 \text{ Nm} \approx 57 \text{ Nm}$$

Use an additional equation to find the overall torque

$$\begin{aligned}\tau_N &= r \times M \times (C_{tr} + g + a) \\ &= 0.127 \times 25 \times (0.303 + 9.81 + 4) \\ &= 44.81 \text{ Nm}\end{aligned}$$

Since, $\tau > \tau_N$ we will consider $\tau = 57 \text{ Nm}$

$\tau = 57 \text{ Nm}$

The calculation we made, were based on assumption. No practical or experimental values were used. This calculation was made only to determine if our assumptions are correct or not. Also to know, using geared motor is feasible or not. From the calculation we got the approximate torque that the motor has to provide is 57 Nm. Then the question arises if it is possible for us to provide the required power to the motor to get 57 Nm torque.

Power Calculation:

$$\begin{aligned}P &= \tau w \\ &= 57 \times 1.11 \\ &= 58.83 \text{ W}\end{aligned}$$

$$w = 1.11 \text{ rad/sec}$$

$$w_0 = v/r$$

$$= 0.7 \text{ rad/sec}$$

$$w = w_0 + \alpha t = 1.11 \text{ } (\alpha = 0.4; t = 1 \text{ sec})$$

$P = 58.83 \text{ W}$

Check: if it is possible to supply 58.83 W power

$$P = VI$$

$$V = 12V$$

$$I = P/V$$

$$= 58.83/12$$

$$= 4.9 \text{ A}$$

$I = 4.9 \text{ A}$

It is possible to supply 5A current. Therefore to supply this power **is possible**.

Then we decided to find out the angular velocity of the robot. We made this calculation to have a clear idea about the robot's speed, which is very much essential for controlling the robot. Another thing we had to keep in mind that, the robot will not have the same speed at every condition.

RPM Calculation:

$$w = 1.11 \text{ rad/sec}$$

$$= 63.5983 \text{ degree/sec}$$

$$= 10.59$$

$$\sim 11 \text{ RPM}$$

$w = 11 \text{ RPM}$

From this calculation, we got a RPM, which will provide a smooth speed to the robot.

Since, we got positive results from all the calculations; we made our decision final to use geared DC motor for our robot. Then we started experiment with DC motors and made our chassis and

wheels. Then we found, some modification in our motor calculation is needed. Like as, the mass of the robot, radius of wheels, volume etc. are changed. Therefore, we got a new value of torque.

Modified Torque Calculation:

Mass of Robot, $M = 45 \text{ kg}$ (modified)

Frictional Coefficient, $\mu = 0.80$ (same as previous)

Acceleration, $a = 4 \text{ ms}^{-2}$ (same as previous)

Gravity, $g = 9.81 \text{ ms}^{-2}$ (same as previous)

Coefficient of Rolling Resistance, $C_{rr} = 0.303 \text{ ms}^{-2}$ (same as previous)

Radius of wheel, $r = 0.122 \text{ m}$ (modified)

Torque Calculation

Torque, $\tau = Fr$

Force calculation:

Stage (1): Starting with acceleration a in a horizontal floor (fig. 3.1)

$$\begin{aligned} F(1) &= F + f_s \\ &= Ma + \mu Mg \\ &= (45 \times 4) + (0.80 \times 45 \times 9.81) \\ &= 533.16 \text{ N} \end{aligned}$$

Stage (2): Climbing 30° Slope with acceleration, a (fig. 3.2)

$$\begin{aligned} F(2) &= F + f_s + F_N \sin \Theta \\ &= Ma + \mu Mg + Mg \sin \Theta \\ &= (45 \times 4) + (0.80 \times 45 \times 9.81) + (45 \times 9.81 \times 0.5) \\ &= 753.885 \text{ N} \end{aligned}$$

Stage (3): In a horizontal floor in a constant velocity ($a=0$) (fig.3.3)

$$\begin{aligned}
 F(3) &= f_s \\
 &= \mu Mg \\
 &= (0.80 \times 45 \times 9.81) \\
 &= 353.16 \text{ N}
 \end{aligned}$$

Table 3.2: Modified torque calculation for different force

Force	Torque, $\tau = Fr$
F (1) = 533.16 N	$\tau (1) = 65.05 \text{ Nm}$
F (2) = 753.885 N	$\tau (2) = 91.97 \text{ Nm}$
F (3) = 353.16 N	$\tau (3) = 43.09 \text{ Nm}$

Observe that, $\tau (3) < \tau (1) < \tau (2)$

Let consider the maximum value of torque.

$$\tau = \tau (2) = 91.97 \text{ Nm} \approx 92 \text{ Nm}$$

Use an additional equation to find the overall torque

$$\begin{aligned}
 \tau_N &= r \times M \times (C_{rr} + g + a) \\
 &= 0.122 \times 45 \times (0.303 + 9.81 + 4) \\
 &= 77.48 \text{ Nm}
 \end{aligned}$$

Since, $\tau > \tau_N$ we will consider $\tau = 92 \text{ Nm}$

$\tau = 92 \text{ Nm}$

For this new torque, power consumption is changed; so, we made the power calculation again and got a new required power.

Power Calculation

$$\begin{aligned}P &= \tau \omega \\&= 23 \times 1.5 \\&= 34.5 \text{ W}\end{aligned}$$

$$\omega_0 = v/r = 1.1 \text{ rad/sec}$$

$$\omega = \omega_0 + \alpha t = 1.5 \text{ } (\alpha=0.4; t= 1 \text{ sec})$$

$$\omega = 1.5 \text{ rad/sec}$$

$$\boxed{P = 34.5 \text{ W}}$$

Power for each motor, $P = 34.5/4 = 8.625 \text{ W}$

$$\boxed{P = 8.625 \text{ W}}$$

Check: if it is possible to supply 8.625 W power

$$P = VI$$

$$V = 12\text{V}$$

$$I = P/V$$

$$= 8,625/12$$

$$= 1 \text{ A}$$

$$\boxed{I = 1 \text{ A}}$$

It is possible to supply 5A current. Therefore to supply this power **is possible**.

Then we made the RPM calculation again.

RPM Calculation:

$$\omega = 1.5 \text{ rad/sec}$$

$$= 85.9 \text{ degree/sec}$$

$$= 86$$

$$\approx 14.33 \text{ RPM}$$

$$\omega = 14.33 \text{ RPM}$$

This RPM also provides a smooth speed to the robot.

3.1.2 Experiment with motor

Finally we include the experimental values of the motor to compare its actual characteristics with our theoretical values. In Practical, we used RPM measuring device to find motors rotation per meter at different current and voltage. We also measure the time of reaching highest RPM at a fixed supplied current and voltage. Then we made the backward calculation to find out the torque.

Table 3.3: Experimental result of geared DC motor

Current, I(A)	Lower Voltage, V _L (V)	Upper Voltage, V _H (V)	Time, t (sec)	RPM	Angular Velocity (Rad/sec)
1.2	3	5	15	42	4.398
1.3	5	7.3	5	51	5.34
1.4	4	8.9	4	65	6.81
1.5	6.5	9.7	4	69	7.2
1.6	7	10.9	5	81	8.48
1.7	7	12	2	90	9.424
1.74	11.5	12	0.5	91.4	9.5713

Using Matlab, we plot all the relative values of Voltages at constant supplied current for having a basic idea about the motor characteristics.

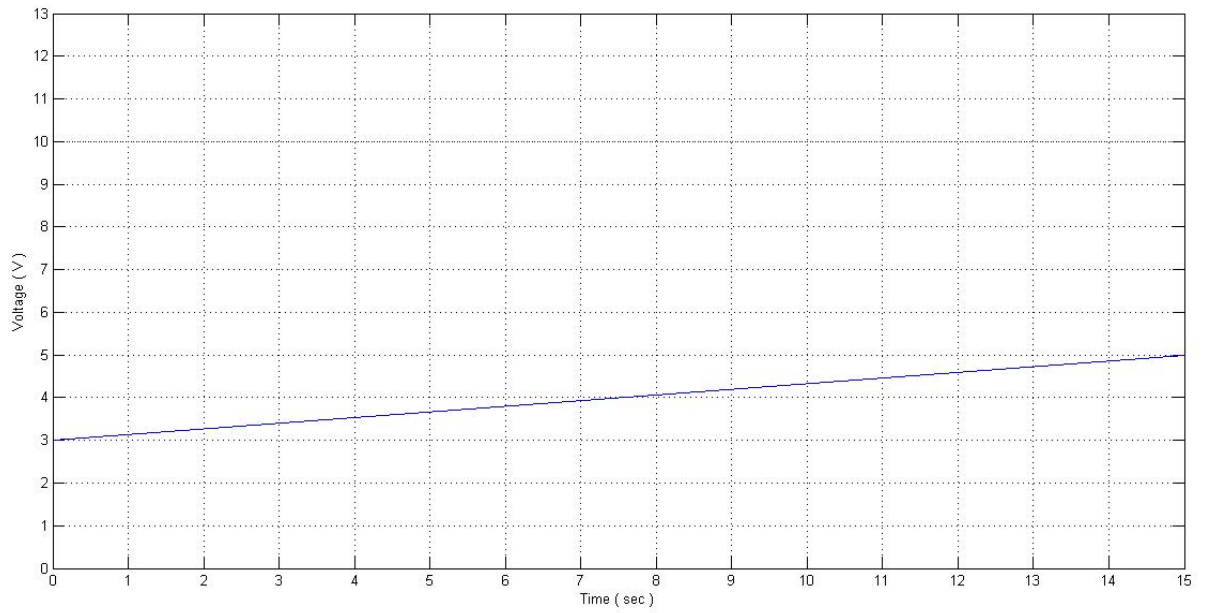


Figure 3.4: Time vs Voltage curve at constant Current, $I= 1.2A$

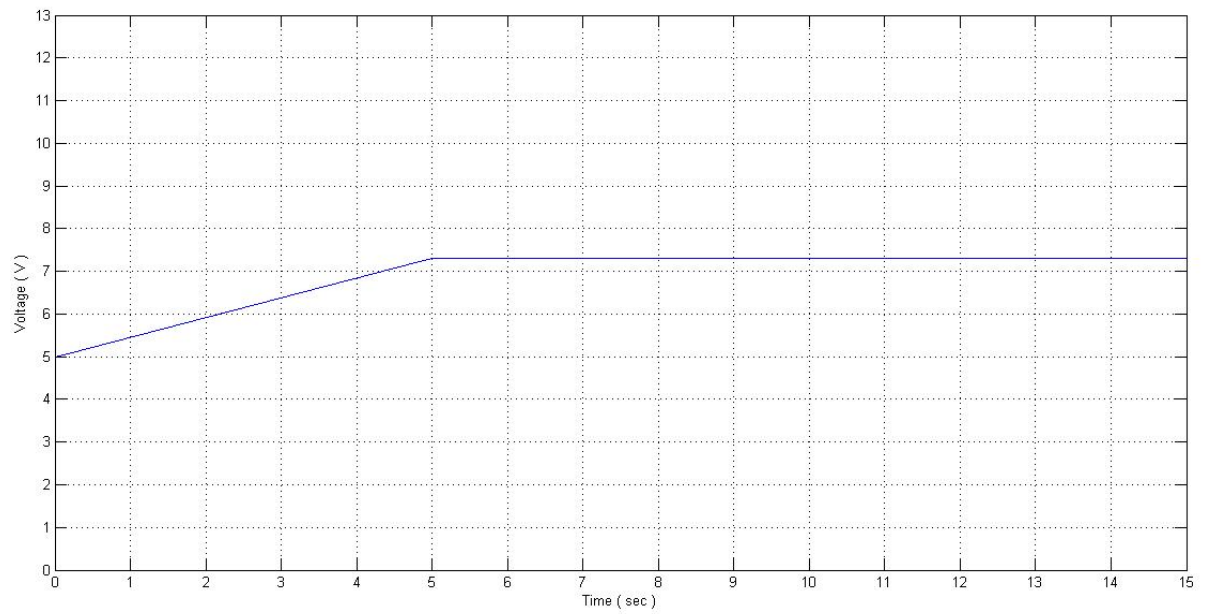


Figure 3.5: Time vs Voltage curve at constant Current, $I= 1.3A$

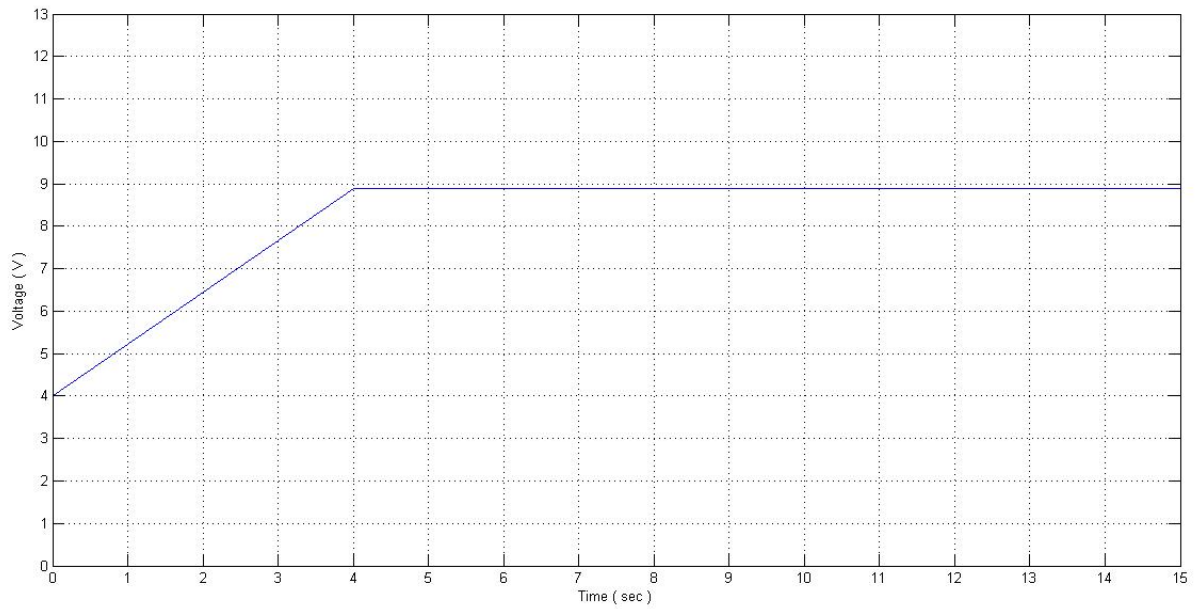


Figure 3.6: Time vs Voltage curve at constant Current, $I= 1.4A$

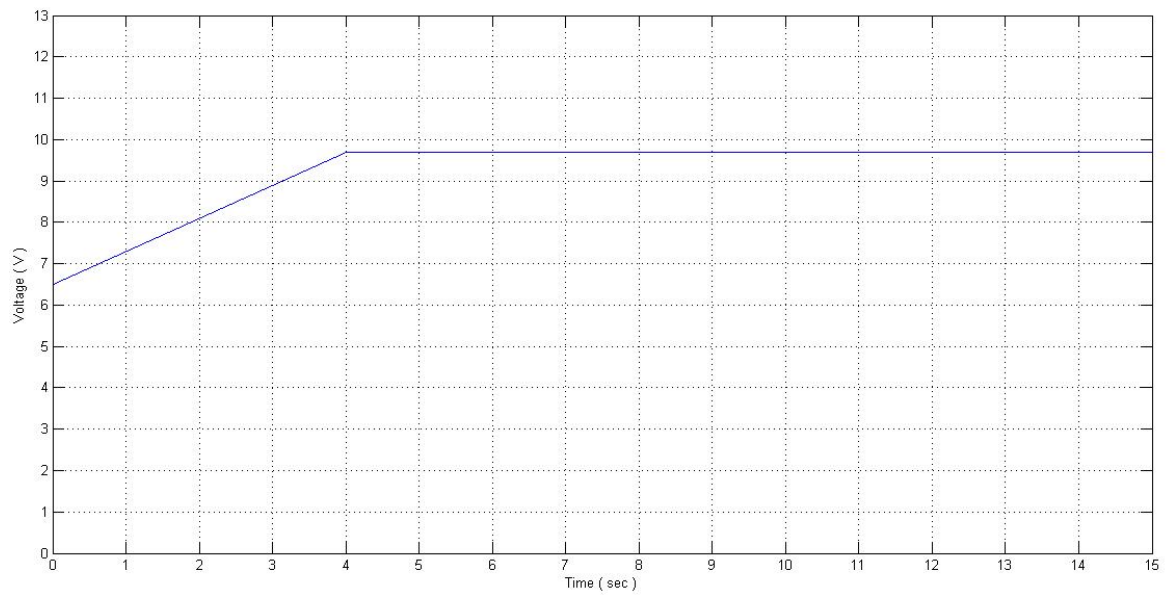


Figure 3.7: Time vs Voltage curve at constant Current, $I= 1.5A$

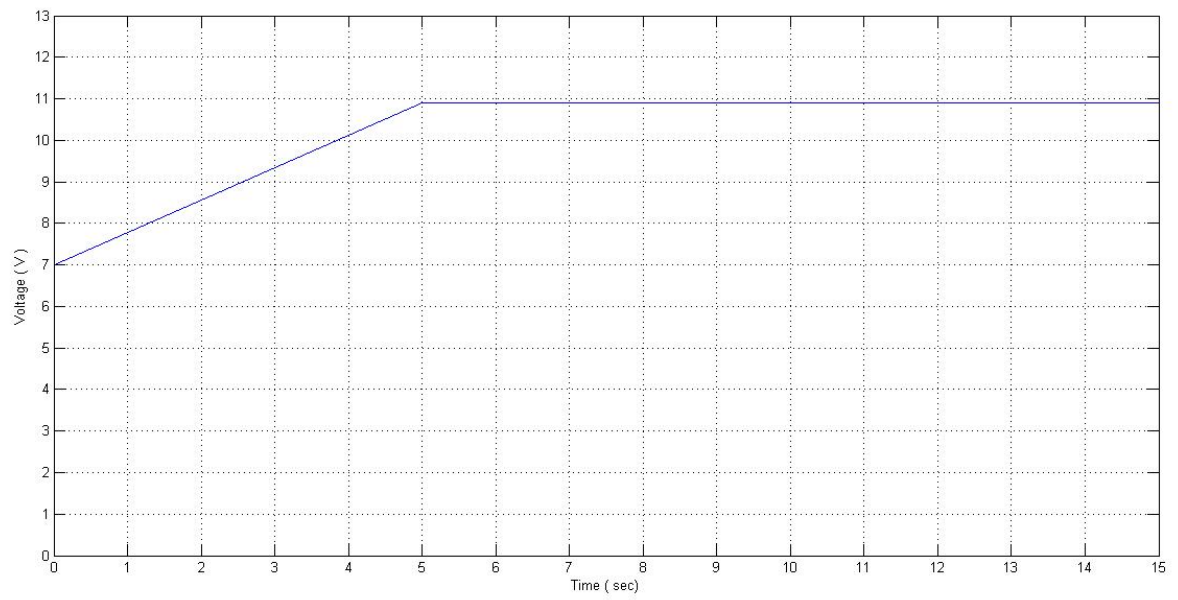


Figure 3.8: Time vs Voltage curve at constant Current, $I= 1.6A$

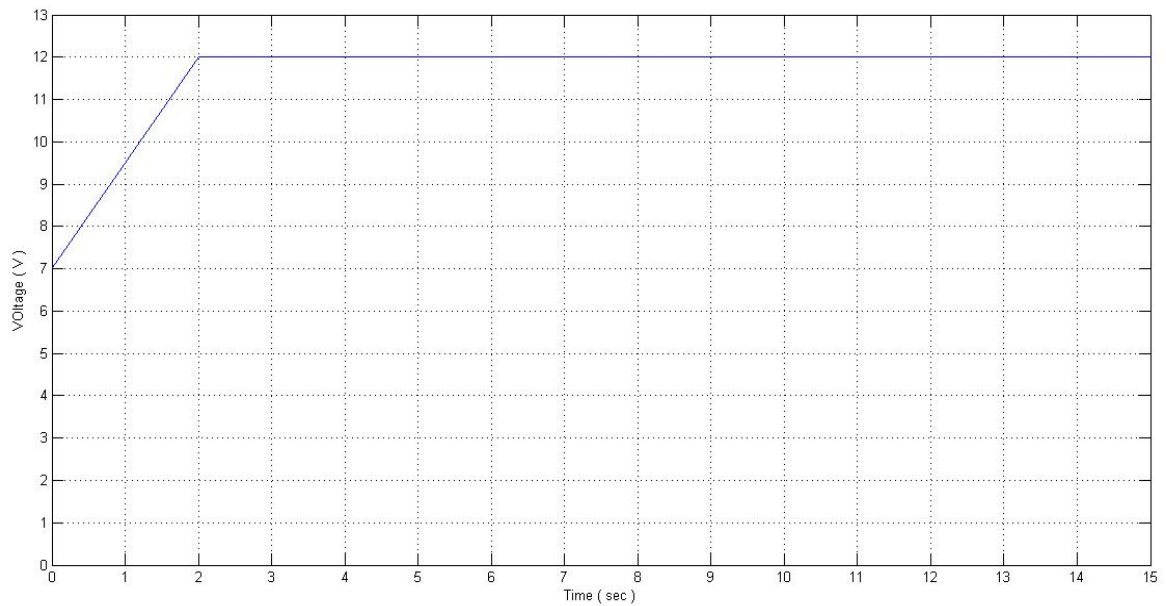


Figure 3.9: Time vs Voltage curve at constant Current, $I= 1.7A$

From the graphs, it is clear that, voltage has a linear relation with current and time. The minimum and maximum voltage is fixed for a constant current. If the supplied current is fixed, with the passing of time, motor's induced voltage increases linearly from the minimum to the maximum value. There is also a minimum current, before which, the motor dose not start. This lowest current is 1.2A. At the very beginning, supplied current is fixed at lowest [Fig. 3.1]. The voltage limit is 3-5V for this current level. Since 1.2A is the lowest current, motor takes maximum time to raise its voltage. At 1.3A, voltage limit is 5-7.3V and time duration is 5 second [Fig.3.2]. With the change of 0.1A current, the time duration of reaching maximum voltage decreases 3 times. After 5 second, the voltage remains same. It will not increase, if the supplied current is not changed. Then increasing the current 0.1A more, we found, the voltage level is not increasing that much like the previous [Fig.3.3]. For 1.4A current, voltage limit is 4-8.9V and time duration is 4 second. However, in this case, the time vs voltage curve is shaper then the previous one. At 1.6A current, the voltage increases up to 10.9 volt within 5 second time [Fig3.5]. Then the maximum current is supplied and the voltage increases to 12V over 2second time. It has the sharpest curve [Fig.3.6]. In the maximum supplied current, the motor moves with its highest rotation and induces maximum voltage. From the graphs it is clear that, the time to induce the maximum voltage decreases if supplied current is increased.

Another experiment was made by supplying unlimited current and voltage to the motor. This was made to measure the motors maximum current commotion, induced voltage and time needed to be started and supply maximum torque. Within 0.5 second time, the induced voltage increases at 12V and the minimum voltage level for this condition was 11.5V [Fig.3.7].

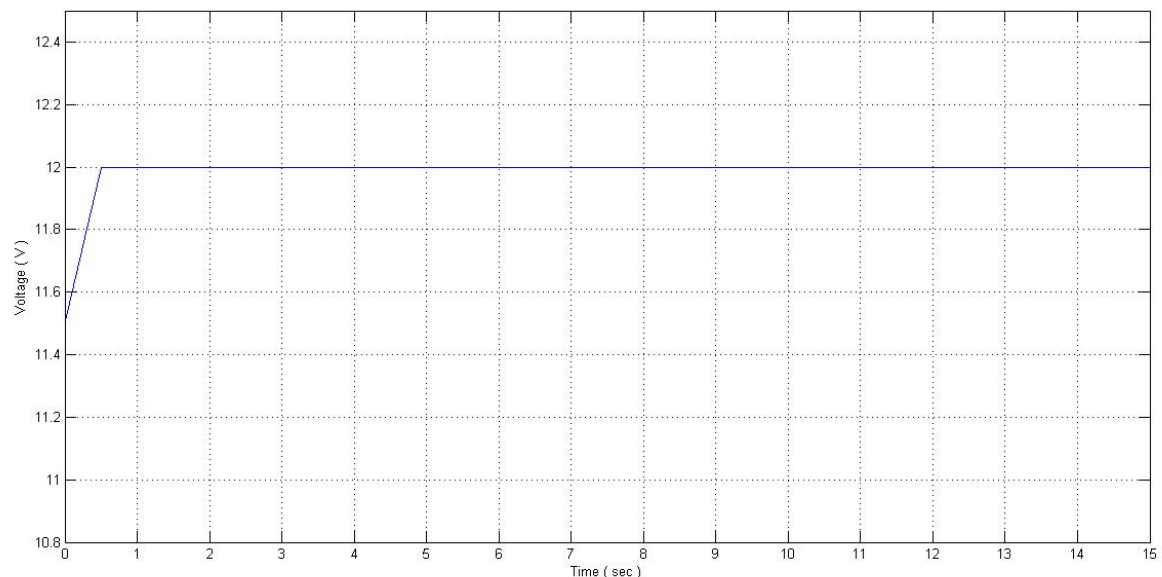


Figure 3.10: Time vs Voltage curve at maximum current

For a geared dc motor, voltage increases linearly with the increase of current. In practical, sometimes it does not work and this relation varies. However, we luckily got an almost linear curve between Current and voltage for our geared motor.

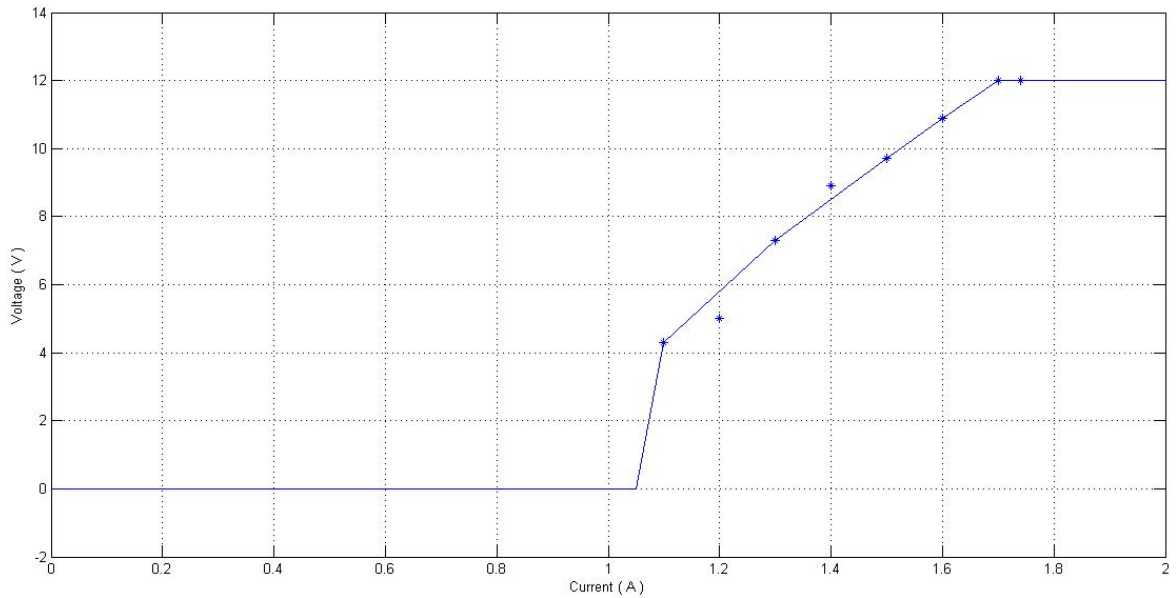


Figure 3.11: Current vs Voltage curve

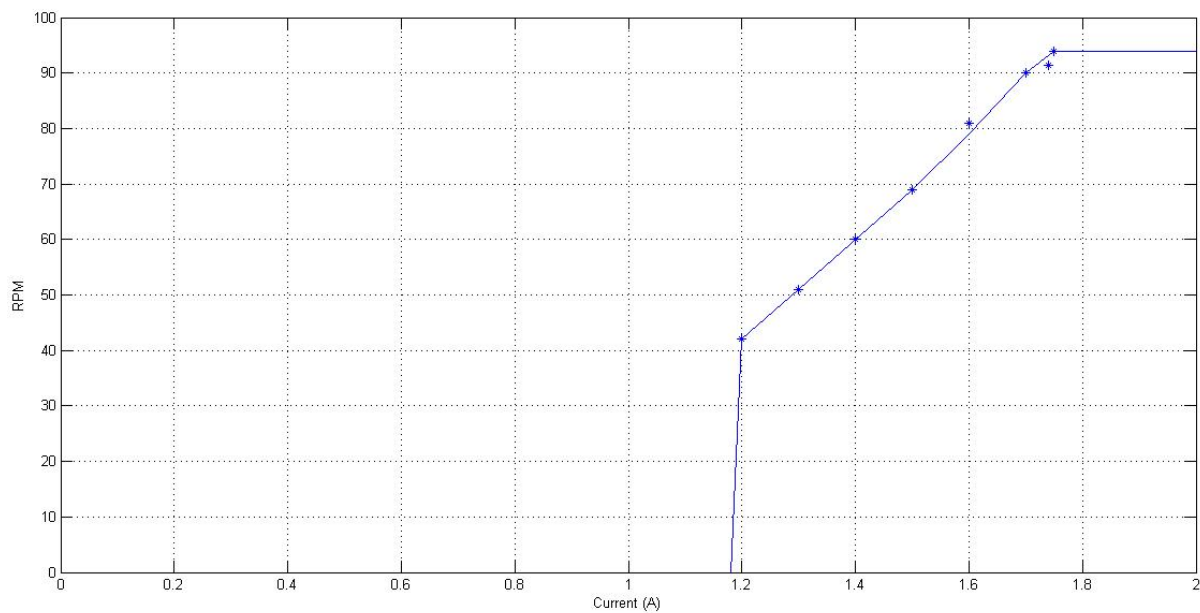


Figure 3.12: Current vs RPM curve

From the curve it is clear that, the motor does not induce any voltage before the current raise to 1.1A. At 1.1A current, there is a sharp change in voltage and it increases to more than 4 volt

within few milliseconds. Then the voltage increases almost linearly up to 12 volt. After that, the voltage is fixed and does not change with the increase of current. Also for a free condition of unlimited current supply, we found, the current does not increase after 2A [Fig.3.8].

Another important thing that we found from our experiment is the relationship between current and RPM of the motor is also linear. We measured the different RPM values at different supplied current and plot them using Matlab [Fig.3.9].

When the current reaches to 1.1A, the motor starts rotating and we get a sharp rise of RPM from 0 to 42. Then it increases linearly with the increase of supplied current up to 91.4. Then this RPM value remains same for the higher values of current. The minimum value of current at which the motor has maximum RPM is 1.74A. Therefore it is enough to supply 2A current for having maximum speed.

3.2 Motor Controlling Circuit:

A permanent magnet DC motor act in response to both voltage and current because the motor's running speed depends on voltage and the current through its armature windings determines the torque [23]. If the power supply does not able to provide enough current, the voltage will drop reducing the speed of the motor whereas if the power supply can uphold voltage while supplying the current, the motor will run at the same speed.

Normally DC gear motor requires more than 250mA current which is unable to supply by any IC or microcontroller. Therefore a circuitry is needed to bridge between the motor and IC/microcontroller. In this research work, for operating the wheels of the fire extinguisher robot selected motor is DC gear motor that is 12V and draws 2A in no load. Also, for moving the robot in 4 directions like forward, backward, left, right motor direction needed to be changed when necessary and so H-bridge circuit is applied for doing so. In dc motor direction of motor rotation depends on voltage difference between terminals and swapping the terminals will cause in opposite rotation of motor [24]. In this case H-bridge circuit let the event occur by changing polarity. Since built-in H-bridge IC i.e. L293D, L298 cannot be employed because of high current requirement of motor, a circuitry combined of N-channel MOSFET with dual pole relay is introduced as motor controlling circuit.

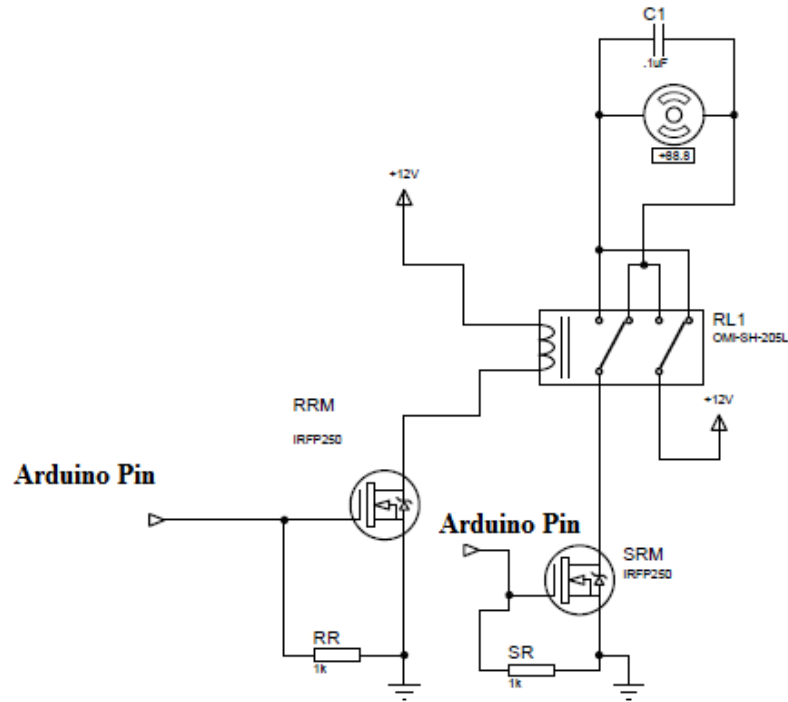


Figure 3.13: Motor Controlling Circuit

The coil rating of dual pole relay is 12V and so to trigger the relay by 5V coming from arduino in between a MOSFET (RRM from fig-3.131) is used as a switch. In this circuit, gate is connected with arduino pin as well as has a pull down resistor, source is grounded and drain is connecting with a part of relay coil. Whenever the gate of MOSFET will receive 5V and $V_{GS} > V_T$ condition fulfills, MOSFET being as switch turned on.

By default relay pins of Common terminal and Normally Close terminal is connected. Whenever a voltage is applied to the relay coil, current flows create magnetic energy which switches the Common terminal from Normally Close Terminal to Normally Open terminal [25]. This principle of relay is employed in this circuitry for changing polarity of motor rotation.

The second MOSFET (SRM from fig-3.13) is responsible for power up the motors. This MOSFET is also acted as a switch where it will complete the circuit every time it turns on and drain-source become shorted thus grounded. Below is the logic table of motor control circuit.

Table 3.4: Logic Table for Motor Control

RRM	SRM	Rotation
0	0	Stop
0	1	Clockwise
1	0	Stop
1	1	Anti-clockwise

3.2.1 Condition of a Single Motor for Each Logic Combination:

3.2.1.1 Logic Combination 00:

For the logic combination of 00, no signal is applied on the gate of the two MOSFETs. According to fig-3.14, because of no voltage applied to gate terminal, no current flows from drain to source because of high resistance of 100MΩ. Even a positive drain voltage cannot be significant in current flowing through the region.

First MOSFET (RRM in fig-3.14) is acting as a trigger for relay and its drain is connected with a terminal of relay coil. Since coil rating is 12V, so 12V is applied and so the whole coil reserve 12V and so drain terminal has a positive voltage of 12V.

The drain terminal of second MOSFET (SRM in fig-3.14) is connected with common terminal of dual pole relay. Since, gate voltage of MOSFET is 0V; Gate-to-Source voltage (V_{GS}) is 0V.

For RRM MOSFET:

$$V_T = 2V \quad [26]$$

$$V_{GS} = V_G - V_S = 0V$$

$$\text{So, } V_{GS} - V_T = -2V$$

3.2.2.2 Logic Combination 01:

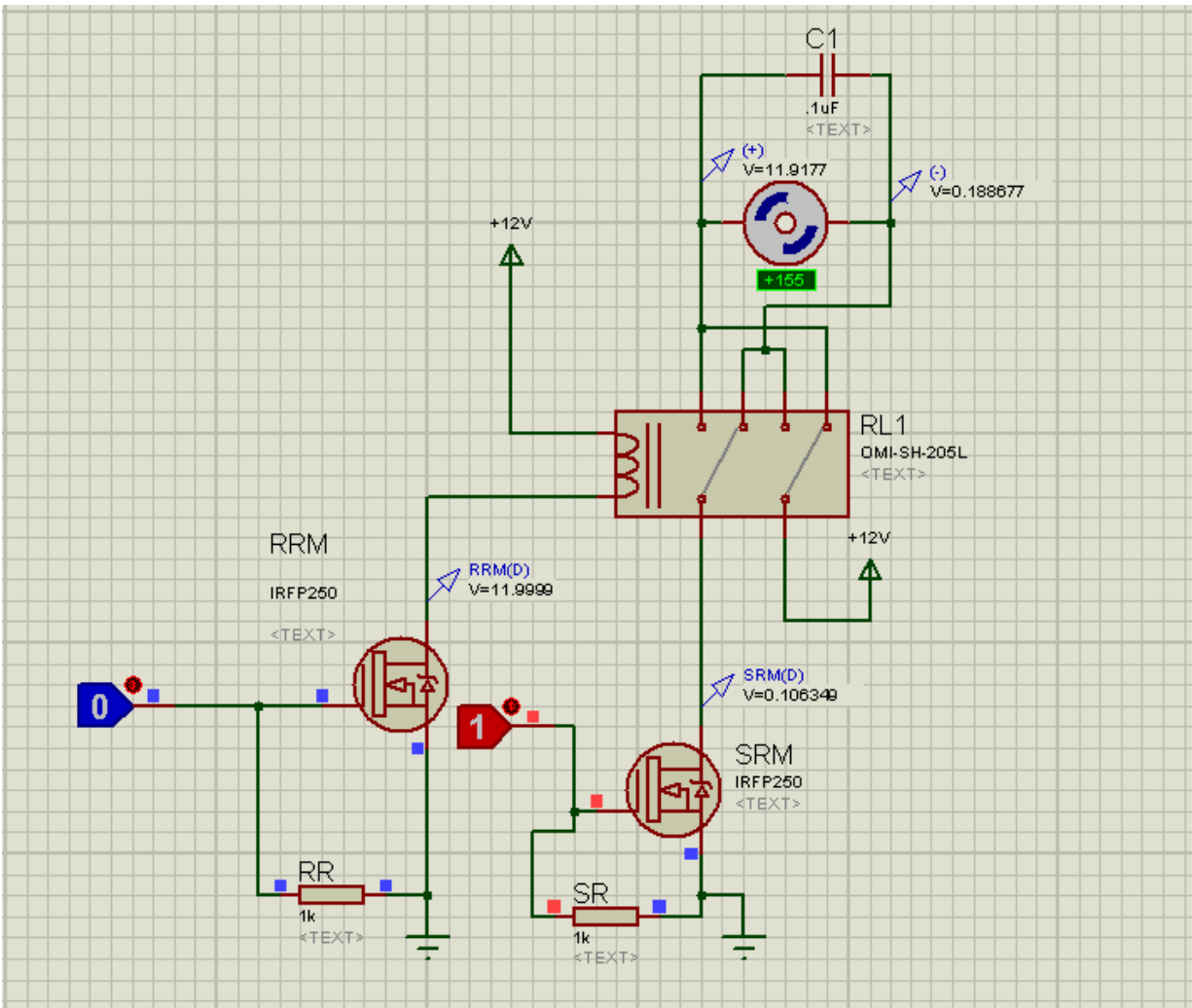


Figure 3.15: Motor rotate clockwise for logic 01

In logic combination of 01, no signal is applied on the gate of RRM MOSFETs which is responsible for changing polarity of motor. Since, no voltage is applied, polarity is not changed and common terminal of relay is connected with normally closed terminal. The drain terminal of that MOSFET is connected with relay coil and holds positive 12V but because of gate voltage 0V no current flows from drain-to-source. According to fig-3.15, motor should rotate clockwise. However, motor run/stop depends on SRM MOSFET.

For RRM MOSFET,

$$V_T = 2V \quad [26]$$

$$V_{GS} = V_G - V_S = 0V$$

$$\text{So, } V_{GS} - V_T = -2V$$

Since $V_{GS} - V_T = -2V \leq 0V$, MOSFET is in cut-off region and it is switched off.

The drain terminal of second MOSFET (SRM in fig-3.15) is connected with common terminal of dual pole relay. However, gate voltage of MOSFET is 5V so Gate-to-Source voltage (V_{GS}) is 5V.

$$V_{GS} = V_G - V_S = 5V$$

Since gate voltage is 5V resistance between drain-to-source terminal lessens to around $10K\Omega$ or less and current flows from drain to source terminals.

$$V_{DS} = V_D - V_S = 0 - 0 = 0V$$

We know that, in linear region [5], $0 < V_{DS} < V_{GS} - V_T$

$$V_{GS} - V_T = 5 - 2 = 3V$$

$$V_{DS} = 0V$$

$$\text{So, } V_{GS} - V_T = 3V > V_{DS}$$

As a result, current flows from drain to source.

3.2.1.3 Logic Combination 10:

According to fig-3.16, arduino mega2560 send logic 1 to RRM MOSFET where SRM MOSFET gets logic 0. Since, RRM MOSFET is responsible for changing polarity of motor direction, logic level 1 or 5V at gate terminal of MOSFET let the opportunity to relay for changing its polarity. Whenever the relay coil gets its rating voltage in this case 12V, current flow creates electromagnetic energy to move the common terminal to switch from NC to NO terminal. Since we are using dual pole relay, there are two NC, NO and CO terminal. One CO terminal is connected with drain terminal of SRM MOSFET where the other CO terminal is connected with the 12V battery. In this situation, CO terminals switch their position and connect to NO terminals, thus swap the terminal of motor.

For RRM MOSFET,

$$V_{GS} = V_G - V_S = 5V$$

$$V_{DS} = V_D - V_S = 0 - 0 = 0V$$

We know that, in linear region [27], $0 < V_{DS} < V_{GS} - V_T$

we are using dual pole relay, there are two NC, NO and CO terminal. One CO terminal is connected with drain terminal of SRM MOSGFET where the other CO terminal is connected with the 12V battery. In this situation, CO terminals switch their position and connect to NO terminals, thus swap the terminal of motor.

For RRM MOSFET,

$$V_{GS} = V_G - V_S = 5V$$

$$V_{DS} = V_D - V_S = 0 - 0 = 0V$$

We know that, in linear region [27], $0 < V_{DS} < V_{GS} - V_T$

$$V_{GS} - V_T = 5 - 2 = 3V$$

$$V_{DS} = 0V$$

$$\text{So, } V_{GS} - V_T = 3V > V_{DS}$$

As a result, current flows from drain to source and V_{DS} becomes 0V and coil voltage becomes 12V.

The drain terminal of second MOSFET which is SRM (SRM in fig-3.17) is connected with common terminal of dual pole relay. However, gate voltage of MOSFET is 5V so Gate-to-Source voltage (V_{GS}) is 5V.

$$V_{GS} = V_G - V_S = 5V$$

Since gate voltage is 5V resistance between drain-to-source terminal lessens to around $10K\Omega$ or less and current flows from drain to source terminals.

$$V_{DS} = V_D - V_S = 0 - 0 = 0V$$

We know that, in linear region [5], $0 < V_{DS} < V_{GS} - V_T$

$$V_{GS} - V_T = 5 - 2 = 3V$$

$$V_{DS} = 0V$$

$$\text{So, } V_{GS} - V_T = 3V > V_{DS}$$

Thus current flows from drain to source and motor will get the power supply to rotate.

3.2.2 Fire Extinguisher Robot Motor Controller Operation:

Motor controller used for this fire extinguisher robot consists of 4 dual pole relays, 8 N channel MOSFETs which means the robot is 4 wheeled. For controlling these motors 8 PWM I/O pins are used. So a combination has to be brought for controlling motors in different directions. Table below represents the functions of different motors in different input.

Table 3.5: Function Table for Fire Extinguisher Robot Control

Motor	Arduino Pin	Applied Logic in RRM	Arduino Pin	Applied Logic in SRM	Rotation	Function
Left Motor	2	0	3	1	Clockwise	Forward
	6	0	9	1	Clockwise	
Right Motor	4	1	5	1	Anti-clockwise	
	10	1	11	1	Anti-clockwise	
Left Motor	2	1	3	1	Anti-clockwise	Backward
	6	1	9	1	Anti-clockwise	
Right Motor	4	0	5	1	Clockwise	
	10	0	11	1	Clockwise	
Left Motor	2	1	3	1	Anti-clockwise	Left
	6	1	9	1	Anti-clockwise	
Right Motor	4	1	5	1	Anti-clockwise	
	10	1	11	1	Anti-clockwise	
Left Motor	2	0	3	1	Clockwise	Right
	6	0	9	1	Clockwise	
Right Motor	4	0	5	1	Clockwise	
	10	0	11	1	Clockwise	

After integrating all the components in our system it has been seen that, for extra layer of protection because of back EMF optocouplers are used in between the Arduino mega2560 and motor controller circuit. In optocoupler transmitter and receiver part are 100% isolated from each other, there is no chance of reverse current flow which protect the microcontroller board from damage.

For speed control of motors PWM techniques are applied from the Arduino PWM pins. This is the reason why PWM pins are selected rather than digital I/O pins. Arduino Mega2560's pins 2 to 13 are for PWM functions. In this study, Pin no. 2,3,4,5,6,9,10,11 are used to do so. Pulse width modulation technique supply electrical power to its load with relative low response and each pulses controls effective voltage level to its load [28]. The fire extinguisher robot is controlled wirelessly from a base station and it is operated by some keyboard input commands. So a dedicated command is set into the hardware of the robot's which will be responsible for carrying out the command.

Chapter -4

Communication System for FireBOT

4.1 Communication System:

An extensive term that incorporates all procedures of connecting more than one device using a high frequency signal is Wireless Communication. A 2.4 GHz Radio Frequency Wireless communication is one of the categories of RF (Radio Frequency) data communication that is carried out and distributed using 2.4 GHz frequency of signal wirelessly through air medium [29]. So, it can be held that it is the transfer of information between two or more points which are not being wired connected. Wireless operations consent services, such as long-range or short-range communications, that are unworkable or impractical to put into practice with the use of wires like communicating with FireBOT (Fire Extinguishing and Surveillance Purpose Robot) in a fire swept place. In such situation, a long range efficient wireless communication could be a convenient solution [30], [31].

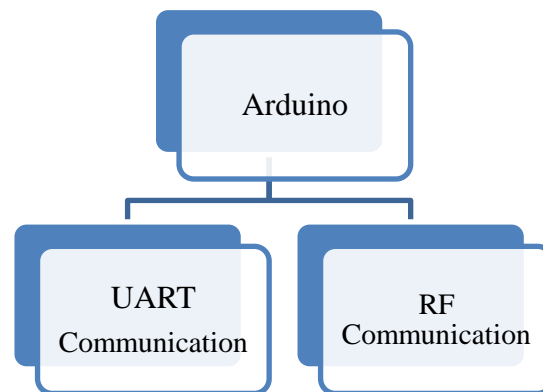


Figure 4.1: Arduino is operated in 2 communication schemes

Arduino is maneuvered in two communication processes. They are:

- ✚ UART Communication
- ✚ RF Communication

Both of the communication will be described below that how they work and how are used for controlling FireBOT Robot.

4.1.1 Serial Communication:

The meaning of Serial is "One subsequent to another". Similarly, the Serial communication is when data is transferred one bit by one bit. By setting a pair of pin of Arduino high or low, information is passed back and forth between the computer and Arduino. The particular software which is developed for controlling FireBOT, can read/send the data from/to the serial port and make further processing of it [32].

Serial Communication in Arduino, uses a serial port also known as UART, which stands for Universal Asynchronous Receiver/Transmitter to Transmit and Receive information. In this case, the computer performs Serial Communication via USB while the Arduino receives and transmits Serial data using the RX and TX pins. This is a very necessary communication which is every time used to upload code to Arduino board and reading data from Arduino [32]. It is also used to debug code and troubleshoot circuits. Basic serial communication is outlined in the following paragraphs along with a simple activity to understand the concepts.

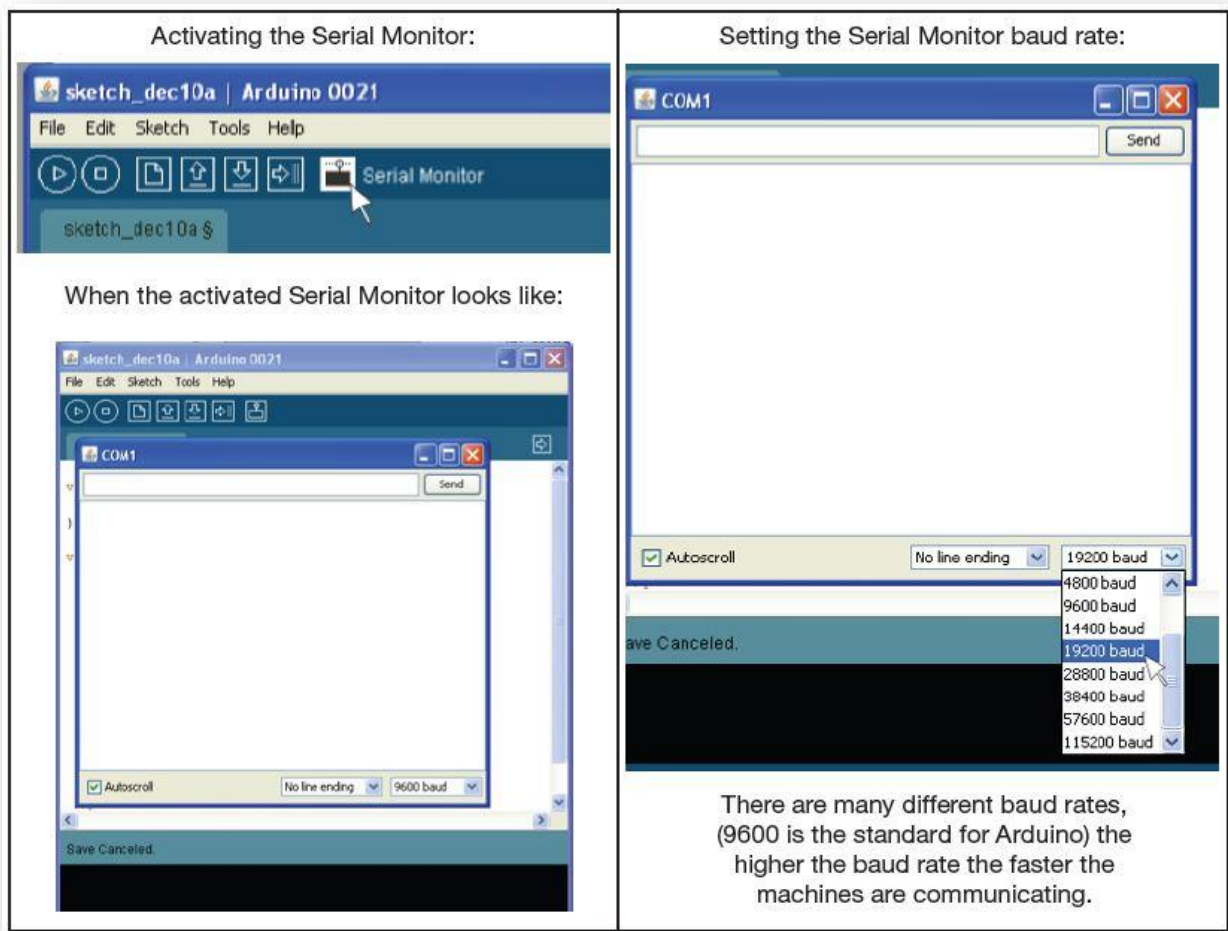


Figure 4.2: Serial Monitor: Monitoring and Setting Up serial Communication Baud Rate

4.1.2 Basic Arduino Coding for Serial Communication:

The first thing it is needed to know to use Serial Communication with Arduino through coding is to do "Serial setup". To setup Serial , it is required to simply type the following line inside setup () function:

`Serial.begin (9600); //Basic Line of Coding in Arduino`

....
....

The aforementioned line establishes that the Digital Pins #0 and #1 on the Arduino are occupied for Serial communication. This means that, it is completely unable to use these pins as Input or Output because they are already dedicated for Serial communication between Arduino and Computer. Here, the number inside bracket "9600" is the baud rate, the rate at which computer and the Arduino communicate. Baud rate can be changed depending on needs but need to make sure that the baud rate in Serial setup and the baud rate on Serial Monitor (Compiler software for Arduino) must be same. If the two baud rates do not match up, the Serial Monitor will display what appears to be garbage which is actually the correct communication but incorrectly translated. So, to setup serial communication between computer and Arduino, any programmer should carefully write the code and put baud rate and also the two baud rates must be similar for accurate reception [33].

4.2 nRF24L01 + LNA + PA Transceiver Communication:

A wireless transceiver RF module is a combination of self-contained circuits with other important electronic components like Amplifiers, Different types of Filters and associated wiring which performs as a Transmitter (TX) and a Receiver (RX) to form a larger system in communication. In this research, an efficient but cost effective nrF24L01+LNA+PA 2.4GHz wireless RF module is used for wireless communication purpose which is intended to supply maximum 1 Kilometer range in open place.



Figure 4.3: nrF24L01 Module in the unconnected state

The foremost concern about communication respectability between the server (From where the Robot will be controlled) and FireBOT is resolved by employing 2.4 GHz GFSK incorporated nrF24L01 wireless module. The choice to use nrF24L01 module was commendable adequate because this remarkable, small fashioned but effective component is a full-duplex communication based Transceiver which is consisted of a transmitter and receiver in a single chip, can receive and transmit data simultaneously in different channels. Moreover, the nrF24L01 is a very functional module for wireless communication because of Power amplifier circuit (PA) & Low Noise Amplifier (LNA), which are used to transmit data in longer solitude place as it is anticipated to be safe from dangerous fire accident place and obviously more resolute performance for other engineering customary.

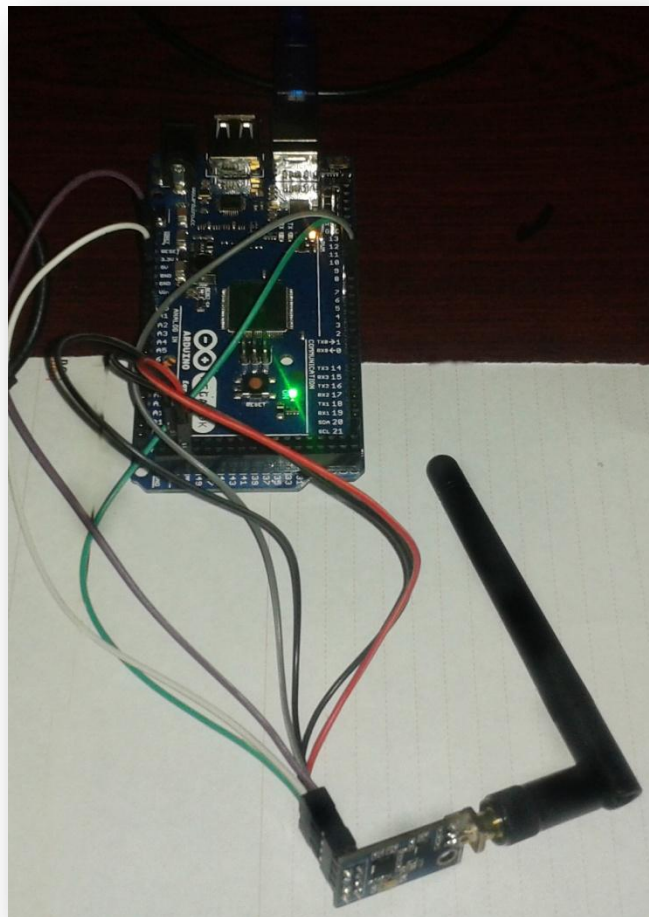


Figure 4.4: nrF24L01 is connected with Arduino (ATmega 2560 Microcontroller)

4.2.1 Reason for choosing nRF24L01+Wireless Transceiver Module:

A professional full dual-way effective communication is always required for ensuring responsiveness of FireBOT which is increased because of PA, LNA, RF switch and Band Pass Filter. A good distance of not less than 1000 meter is covered in LOS (Light of Sight) condition of these wireless modules which is verified by our experiment with FireBOT. These smooth communications are happened between server and FireBOT because of 2dB Antenna at 250kbps data rate while the FireBOT is in action to receive command from serer and make it ready for any response [34].

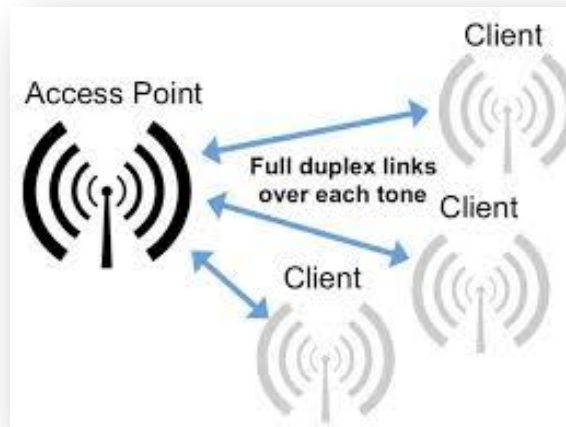


Figure 4.5: General Wireless Communication among Access point and clients

In our research, the Access Point is similar to Server from where FireBOT will be controlled and on the other hand, Client is reminiscent of the FireBOT itself. In the both sides, nRF24L01+'s Transceiver is used for establishing an uninterruptible wireless communication.

4.3 Functional Block Diagrams of nRF24L01+:

There are 14 different functional blocks of nRF24L01+ module which is depicted below:

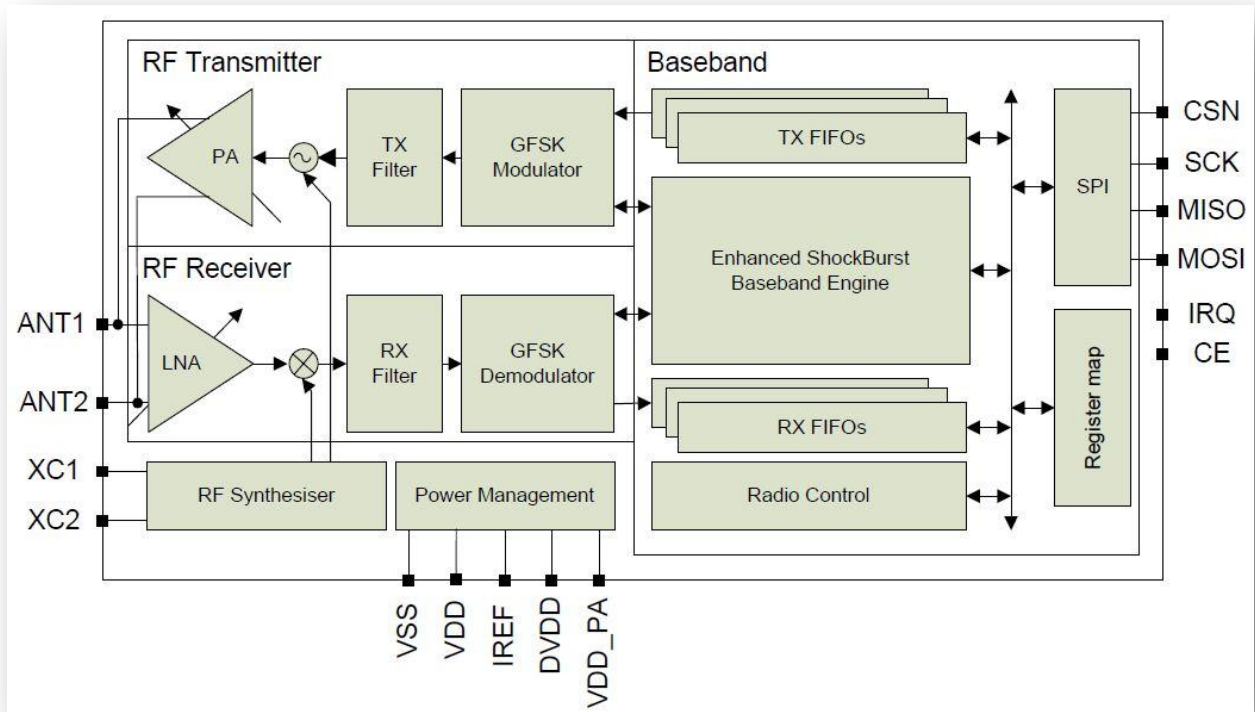


Figure 4.6: nRF24L01+ module's internal circuitry block diagram

The functions of different blocks in this module depend on circuit design. As communication plays a very important role in making of FireBOT so it was required to know about different functionalities of nrf24L01+ module. The main circuit block diagram for nRF24L01 wireless module is divided into three parts. Including those important block diagrams, other electronic components are also given below:

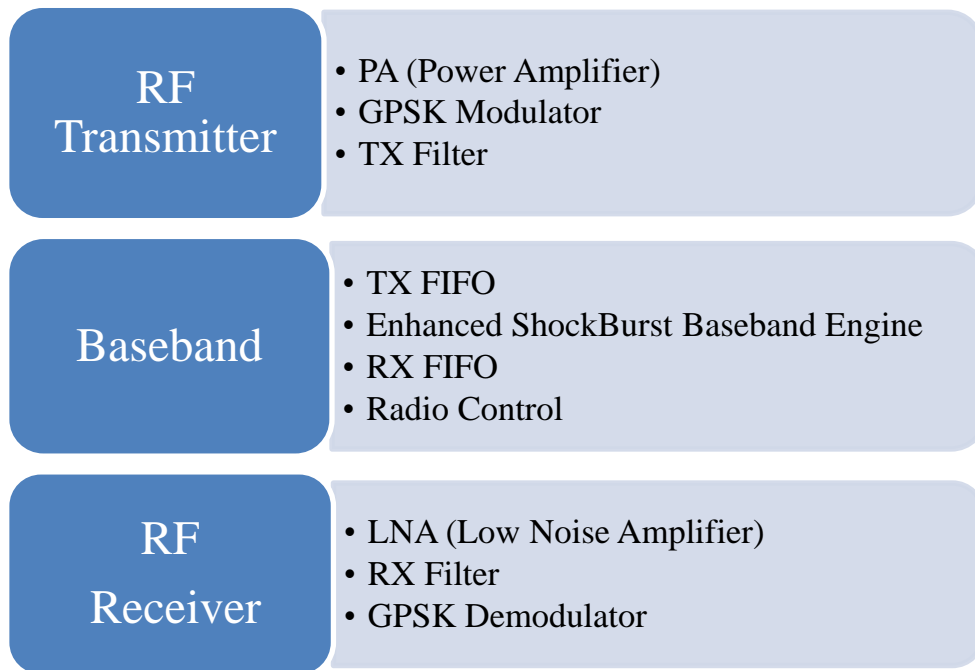


Figure 4.7: Expansion of nRF24L01+ Wireless Module's major Three electronic components

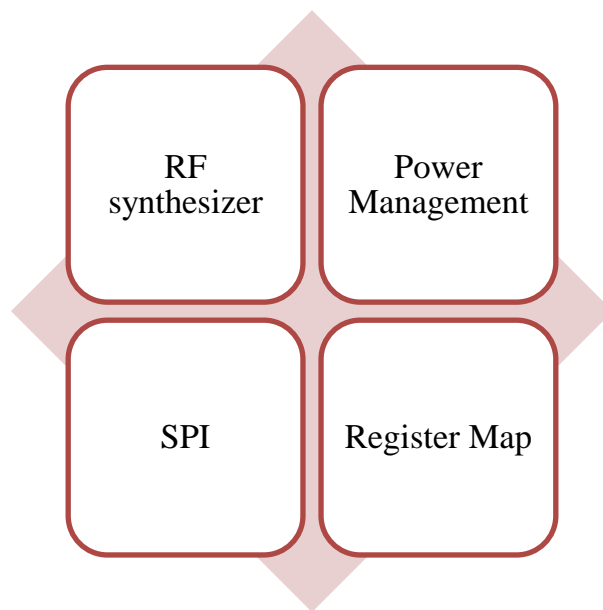


Figure 4.8: nRF24L01+ Wireless Module's other Four important electronic components

Individual block diagram's explanation is given in the following paragraphs.

4.3.1 PA - (Power Amplifier)

An RF power amplifier is a type of electronic amplifier used to convert a low-power radio-frequency signal into a larger signal of significant power, typically for driving the antenna of a transmitter. It is usually optimized to have high efficiency, high output Power, good return loss on the input and output, good gain and optimum heat dissipation.

The basic applications of the RF power amplifier include driving to another high power source, driving a transmitting antenna and exciting microwave cavity resonators. Among these applications, driving transmitter antennas is most well known. In this nrF24L01+, power amplifier is used to amplify the data which are given to send through air.

4.3.2 Transmission (TX) Filter:

Electronic filters are analog circuits which perform signal processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones, or both. Electronic filters can be:

- ✚ Passive or Active
- ✚ Analog or Digital
- ✚ High-pass, Low-pass, Band Pass, Band-reject (band reject; notch), or All-pass.
- ✚ Discrete-time (sampled) or Continuous-time
- ✚ Linear or Non-linear
- ✚ Infinite Impulse Response (IIR type) or Finite Impulse Response (FIR type)

The most common types of electronic filters are linear filters, regardless of other aspects of their design. The nrF24L01 uses Digital filtering for Transmission. Digital filtering has specific characteristics that needs to pay special attention. The analog input signal must satisfy certain requirements.

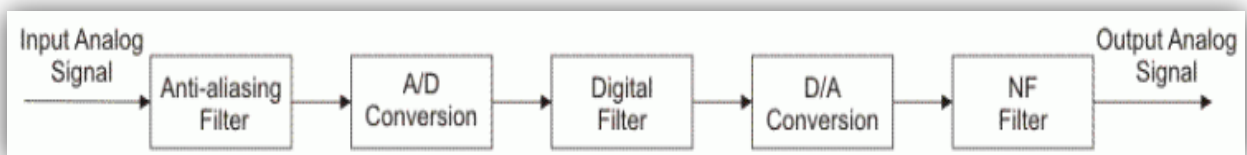


Figure 4.9: Digital Filtering block diagram

Furthermore, on converting an output digital signal into analog form, it is necessary to perform additional signal processing in order to obtain the appropriate result which is done in nrF24L01 RF module. See the article on linear filters for details on their design and analysis [35], [36].

4.3.3 GPSK modulator:

Phase-shift keying (PSK) is a digital modulation scheme that conveys data by changing, or modulating, the phase of a reference signal (the carrier wave). GPSK is a type of Phase Shift Keying which refers to Gaussian PSK. Any digital modulation scheme uses a finite number of distinct signals to represent digital data. It uses a finite number of phases and each phase is assigned with a unique pattern of binary digits. Usually, each phase encodes an equal number of bits. Each pattern of bits forms the symbol that is represented by the particular phase. The demodulator, which is designed specifically for the symbol-set used by the modulator, determines the phase of the received signal and maps it back to the symbol it represents, thus recovering the original data. In this case, nRF14L01 works as a demodulator and modulator. [37]

4.3.4 TX FIFO:

FIFO is an acronym for "First In, First Out", a method for organizing and manipulating a data buffer, or data stack, where the oldest entry, or 'bottom' of the stack, is processed first. It is analogous to processing a queue with first-come, first-served (FCFS) behavior: where the people leave the queue in the order in which they arrive [38]

FIFO full/empty:

A hardware FIFO is used for synchronization purposes. It is often implemented as a circular queue and thus has two pointers:

1. Read Pointer/Read Address Register
2. Write Pointer/Write Address Register

Read and write addresses are initially both at the first memory location and the FIFO queue is Empty.

FIFO Empty

When the read address register reaches the write address register, the FIFO triggers the Empty signal.

FIFO FULL

When the write address register reaches the read address register, the FIFO triggers the FULL signal.

In both cases FIFO empty and FIFO full, the read and write addresses must be equal. To distinguish between the two situations, a simple and robust solution is to add one extra bit for each read and write address which is inverted each time the address wraps. With this set up, the conditions are:

FIFO Empty

When the read address register equals the write address register, the FIFO is empty.

FIFO FULL

When the read address LSBs equal the write address LSBs and the extra MSBs are different, the FIFO is full.

4.3.5 LNA - Low noise amplifier:

Low-noise amplifier (LNA) is an electronic amplifier used to amplify possibly very weak signals (for example, captured by an antenna). It is usually located very close to the detection device to reduce losses in the feedline. This active antenna arrangement is frequently used in microwave systems like GPS, because coaxial cable feedline is very lossy at microwave frequencies, e.g. a loss of 10% coming from few meters of cable would cause a 10% degradation of the signal-to-noise ratio (SNR) [39].

4.3.6 RF synthesizer:

A frequency synthesizer is an electronic system for generating any of a range of frequencies from a single fixed time base or oscillator. They are found in many modern devices, including radio receivers, mobile telephones, radiotelephones, walkie-talkies, CB radios, satellite receivers, GPS systems, etc. A frequency synthesizer can combine frequency multiplication, frequency division, and frequency mixing (the frequency mixing process generates sum and difference frequencies) operations to produce the desired output signal.

A phase locked loop is a feedback control system. It compares the phases of two input signals and produces an error signal that is proportional to the difference between their phases. The error signal is then low pass filtered and used to drive a voltage-controlled oscillator (VCO) which creates an output frequency. The output frequency is fed through a frequency divider back to the input of the system, producing a negative feedback loop [40]. If the output frequency drifts, the phase error signal will increase, driving the frequency in the opposite direction so as to reduce the error. Thus the output is locked to the frequency at the other input. This other input is called the reference and is usually derived from a crystal oscillator, which is very stable in frequency.

The block diagram below shows the basic elements and arrangement of a PLL based frequency synthesizer.

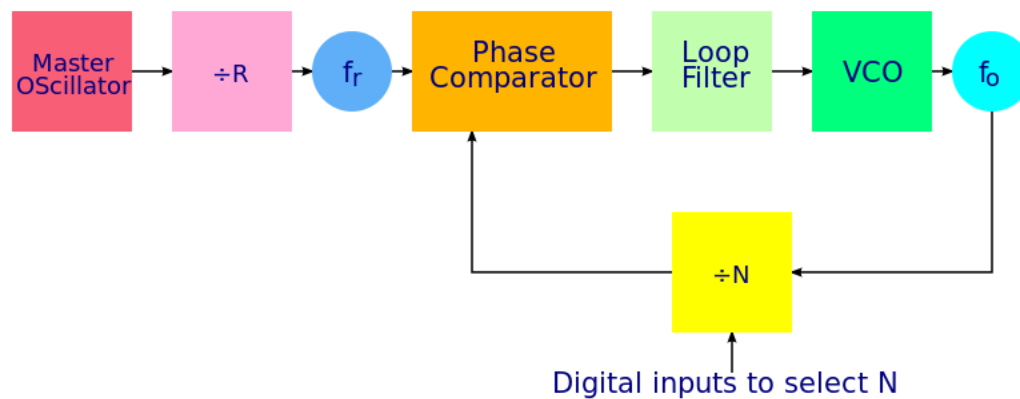


Figure 4.10: RF synthesizer Block diagram

The key to the ability of a frequency synthesizer to generate multiple frequencies is the divider placed between the output and the feedback input. This is usually in the form of a digital counter, with the output signal acting as a clock signal. The counter is preset to some initial count value, and counts down at each cycle of the clock signal. When it reaches zero, the counter output changes state and the count value is reloaded. This circuit is straightforward to implement using flip-flops, and because it is digital in nature, is very easy to interface to other digital components or a microprocessor. This allows the frequency output by the synthesizer to be easily controlled by a digital system [41].

4.4 Details of nRF24L01+ Wireless Module:

The nRF24L01+ is a single chip 2.4GHz transceiver with an embedded baseband protocol engine (Enhanced ShockBurst), suitable for ultra low power wireless applications. The nRF24L01+ is designed for operation in the world wide ISM frequency band at 2.400 - 2.4835GHz. To design a radio system with the nRF24L01+, simply an MCU (microcontroller) and a few external passive components are required.

The nRF24L01+ is configurable through a Serial Peripheral Interface (SPI). The register map, which is accessible through the SPI, contains all configuration registers in the nRF24L01+ and is accessible in all operation modes of the chip. [42]

The embedded Baseband Protocol Engine, which is the heart of this module is based on packet communication and supports various modes from manual operation to advanced autonomous protocol operation. Internal FIFOs ensure a smooth data flow between the radio front end and the system's MCU. The radio front end uses GFSK modulation. It has user configurable parameters like frequency channel, output power and air data rate. The nRF24L01+ supports an air data rate of 250 kbps, 1 Mbps and 2Mbps. The addition of internal filtering to nRF24L01+ has improved the margins for meeting RF regulatory standards. Internal voltage regulators ensure a high Power Supply Rejection Ratio (PSRR) and a wide power supply range and these are the reasons for which to make an effective communication system for FireBOT we have chosen this sophisticated wireless module .

4.4.1 Radio Control:

The nRF24L01+ radio transceiver's operating modes and the parameters are used to control the radio. The nRF24L01+ has a built-in state machine that controls the transitions between the chip's operating modes. The state machine takes input from user defined register values and internal signals.

4.4.2 State diagram:

A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction. Many forms of state diagrams exist, which differ slightly and have different semantics.

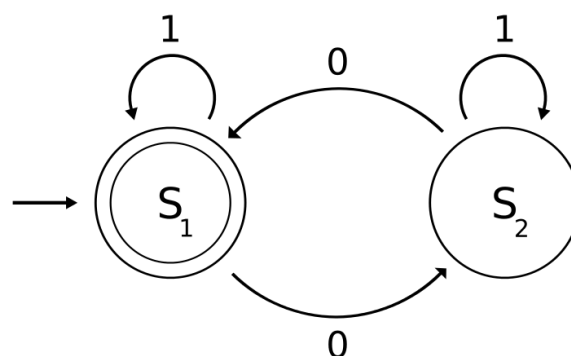


Figure 4.11: An example of a state diagram

S_1 and S_2 are states and S_1 is an accepting state or a final state. Each edge is labeled with the input. This example shows an acceptor for strings over $\{0, 1\}$ that contain an even number of zeros.

The state diagram in Figure 4.12 shows the operating modes of NRF24L01+ and how they function. There are three types of distinct states highlighted in the state diagram:

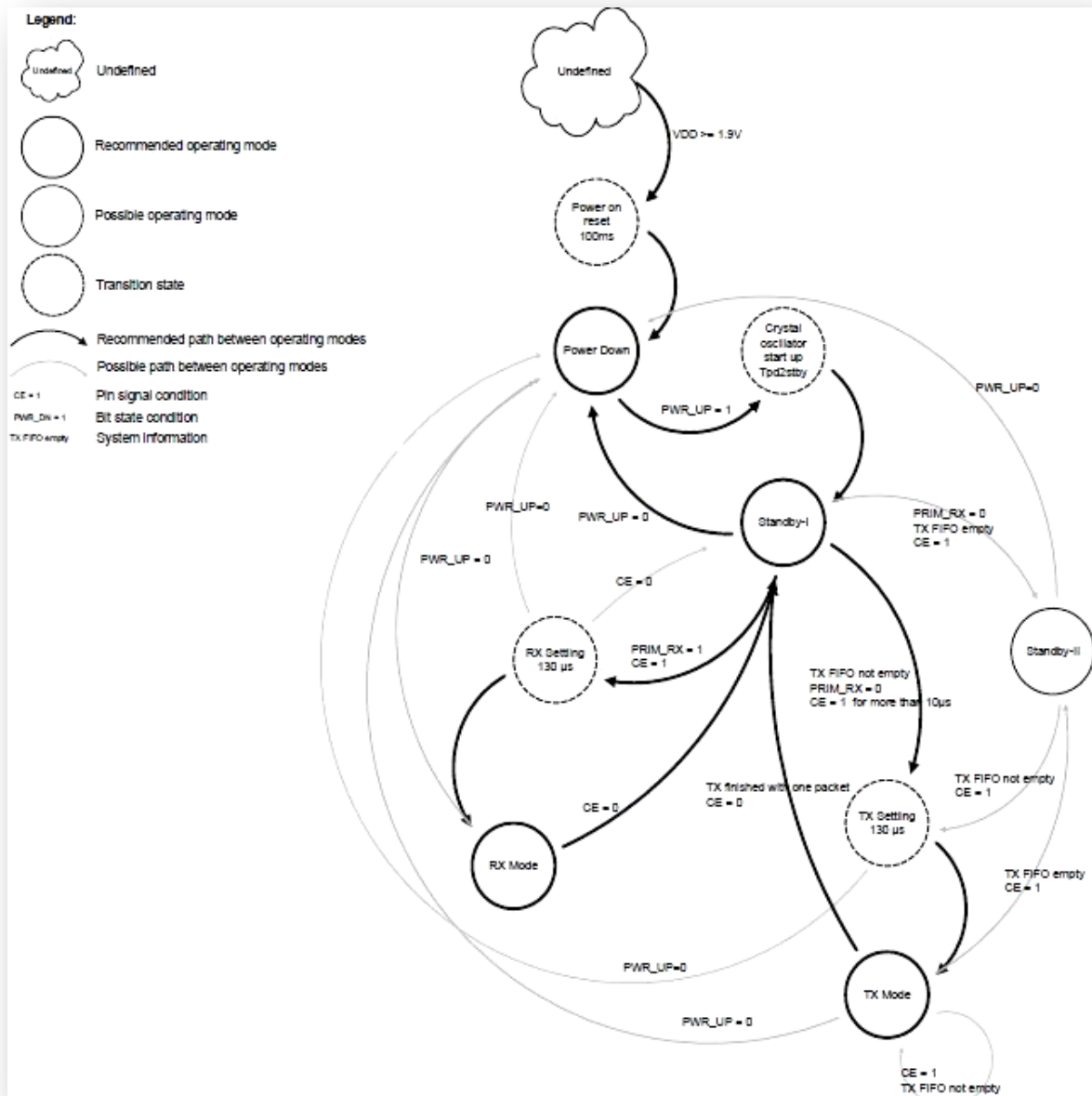





Figure 4.12: State diagram of nRF24L01+ Module

4.4.3 Operational Modes of nRF24L01+:

The nRF24L01+ can be configured in Power Down, Standby, RX or TX mode. Below describes these modes in detail. Before going to different modes, State Diagram of nRF24L01+ is depicted below that creates a clear picture of entire functionalities of this module.

-  Recommended operating mode: It is a recommended state used during normal operation.
-  Possible operating mode: It is a possible operating state, but is not used during normal operation.
-  Transition state: It is a time limited state used during start up of the oscillator and settling of the PLL. When the VDD reaches 1.9V or higher nRF24L01+ enters the Power on reset state where it remains in reset until entering the Power Down mode.

4.4.3.1 Power Down Mode

In power down mode nRF24L01+ is disabled using minimal current consumption. All register values available are maintained and the SPI is kept active, enabling change of configuration and the uploading/downloading of data registers [43].

4.4.3.2 Standby Mode

Standby-I mode:

By setting the PWR_UP bit in the CONFIG register to 1, the device enters standby-I mode. Standby-I mode is used to minimize average current consumption while maintaining short start up times. In this mode only part of the crystal oscillator is active. Change to active modes only happens if CE is set high and when CE is set low, the nRF24L01+ returns to standby-I mode from both the TX and RX modes.

Standby-II mode:

In standby-II mode extra clock buffers are active and more current is used compared to standby-I mode. The nRF24L01+ enters standby-II mode if CE is held high on a PTX device with an empty TX FIFO. If a new packet is uploaded to the TX FIFO, the PLL immediately starts and the packet is transmitted after the normal PLL settling delay (130µs). Register values are maintained and the SPI can be activated during both standby modes [43].

4.4.3.3 Receiving mode:

The RX mode is an active mode where the nRF24L01+ radio is used as a receiver. To enter this mode, the nRF24L01+ must have the PWR_UP bit, PRIM_RX bit and the CE pin set high. In RX mode the receiver demodulates the signals from the RF channel, constantly presenting the demodulated data to the baseband protocol engine. The baseband protocol engine constantly searches for a valid packet. If a valid packet is found (by a matching address and a valid CRC) the payload of the packet is presented in a vacant slot in the RX FIFOs. If the RX FIFOs are full, the received packet is discarded. The nRF24L01+ remains in RX mode until the MCU configures it to standby-I mode or power down mode. However, if the automatic protocol features in the baseband protocol engine are enabled, the nRF24L01+ can enter other modes in order to execute the protocol. In RX mode a Received Power Detector (RPD) signal is available. The RPD is a signal that is set high when a RF signal higher than -64 dBm is detected inside the receiving frequency channel. The internal RPD signal is filtered before presented to the RPD register.

4.4.3.4 Transmission mode:

The TX mode is an active mode for transmitting packets. To enter this mode, the nRF24L01+ must have the PWR_UP bit set high, PRIM_RX bit set low, a payload in the TX FIFO and a high pulse on the CE for more than 10µs. The nRF24L01+ stays in TX mode until it finishes transmitting a packet. If CE = 0, nRF24L01+ returns to standby-I mode. If CE = 1, the status of the TX FIFO determines the next action. If the TX FIFO is not empty the nRF24L01+ remains in TX mode and transmits the next packet. If the TX FIFO is empty the nRF24L01+ goes into standby-II mode. The nRF24L01+ transmitter PLL operates in open loop when in TX mode. It is important never to keep the nRF24L01+ in TX mode for more than 4ms at a time.

4.5 Data Tempo of nRF24L01+:

The air data rate is the modulated signaling rate the nRF24L01+ uses when transmitting and receiving data. It can be 250kbps, 1Mbps or 2Mbps. Using lower air data rate gives better receiver sensitivity than higher air data rate. But, high air data rate gives lower average current consumption and reduced probability of on-air collisions. The air data rate is set by the RF_DR bit in the RF_SETUP register. A transmitter and a receiver must be programmed with the same air data rate to communicate with each other. nRF24L01+ is fully compatible with nRF24L01. For compatibility with nRF2401A, nRF2402, nRF24E1, and nRF24E2 the air data rate must be set to 250kbps or 1Mbps.

4.6. RF channel frequency:

The RF channel frequency determines the center of the channel used by the nRF24L01+. The channel occupies a bandwidth of less than 1MHz at 250kbps and 1Mbps and a bandwidth of less than 2MHz at 2Mbps. nRF24L01+ can operate on frequencies from 2.400GHz to 2.525GHz. The programming resolution of the RF channel frequency setting is 1MHz. At 2Mbps the channel occupies a bandwidth wider than the resolution of the RF channel frequency setting. To ensure non-overlapping channels in 2Mbps mode, the channel spacing must be 2MHz or more. At 1Mbps and 250kbps the channel bandwidth is the same or lowers than the resolution of the RF frequency.

The RF channel frequency is set by the RF_CH register according to the following formula:

$$F0 = 2400 + RF_CH [MHz]$$

4.7 PA (Power Amplification) control:

The PA (Power Amplifier) control is used to set the output power from the nRF24L01+ power amplifier. In TX mode PA control has four programmable steps, see Table 17. The PA control is set by the RF_PWR bits in the RF_SETUP register.

Conditions: VDD = 3.0V, VSS = 0V, TA = 27°C, Load impedance = 15 Ω +j88 Ω.

SPI RF-SETUP (RF_PWR)	RF output power	DC current consumption
11	0dBm	11.3mA
10	-6dBm	9.0mA
01	-12dBm	7.5mA
00	-18dBm	7.0mA

Figure 4.13: RF output power setting for the nRF24L01+

4.8 RX/TX control:

The RX/TX control is set by PRIM_RX bit in the CONFIG register and sets the nRF24L01+ in transmit / receive mode.

4.9 Enhanced ShockBurst:

Enhanced ShockBurst is a packet based data link layer that features automatic packet assembly and timing, automatic acknowledgement and retransmissions of packets. It has made the nRF24L01+ as a unique tool for communication. The Enhanced ShockBurst enables the implementation of ultra low power and high performance communication with low cost host microcontrollers. Enhanced ShockBurst features enable significant improvements of power efficiency for bidirectional and uni-directional systems, without adding complexity on the host controller side.

4.9.1 Enhanced ShockBurst overview:

Enhanced ShockBurst uses ShockBurst for automatic packet handling and timing. During transmit, ShockBurst assembles the packet and clocks the bits in the data packet for transmission. During receive, ShockBurst constantly searches for a valid address in the demodulated signal. When Shock- Burst finds a valid address, it processes the rest of the packet and validates it by CRC. If the packet is valid the payload is moved into a vacant slot in the RX FIFOs. All high speed bit handling and timing is controlled by ShockBurst. Enhanced ShockBurst features automatic packet transaction handling for the easy implementation of a reliable bi-directional data link. An Enhanced ShockBurst packet transaction is a packet exchange between two transceivers, with one transceiver acting as the Primary Receiver (PRX) and the other transceiver acting as the Primary Transmitter (PTX). An Enhanced ShockBurst packet transaction is always initiated by a packet transmission from the PTX, the transaction is complete when the PTX has received an acknowledgment packet (ACK packet) from the PRX. The PRX can attach user data to the ACK packet enabling a bi-directional data link [43].

4.9.2 Enhanced Shockburst packet format:

The format of the Enhanced ShockBurst packet is described in this section. The Enhanced Shock- Burst packet contains a preamble, address, packet control, payload and CRC field.

Preamble 1 byte	Address 3-5 byte	Packet Control Field 9 bit	Payload 0 - 32 byte	CRC 1-2 byte
-----------------	------------------	----------------------------	---------------------	--------------

Figure 4.14: Packet format of nRF24L01+ with MSB to the left.

4.10. Arduino Mega and nRF24L01+ Communication Process:

Pin Func	GND	3.3V/Vdd	SCS	SCK	MOSI	MISO
nRF24L01+PA+LNA	1 (GND)	2.VCC& 3.CE	4 (CSN)	5 (SCLK)	6 MOSI	7 MISO

Figure 4.15: Pin connection of nRF24L01+ Wireless Module

The above table shows general idea about functions of Seven (7) pins in nRF24L01+ Module.

4.10.1 Serial Peripheral Interface:

SPI is an acronym for “Serial Peripheral Interface”. It is a synchronous serial data bus – data can travel in both directions at the same time, as opposed to (for example) the I2C bus that cannot do so. To allow synchronous data transmission, the SPI bus uses four wires. They are called: [43]

- ✚ MOSI – Master-out, Slave-in. This line carries data from our Arduino to the SPI- controlled device(s);
- ✚ MISO – Master-in, Slave out. This line carries data from the SPI-controlled device(s) back to the Arduino;
- ✚ SS – Slave-select. This line tells the device on the bus we wish to communicate with it. Each SPI device needs a unique SS line back to the Arduino;
- ✚ SCK – Serial clock.

For Arduino Mega– MISO is 50, MOSI is 51, SCK is 52 and SS is usually 53.

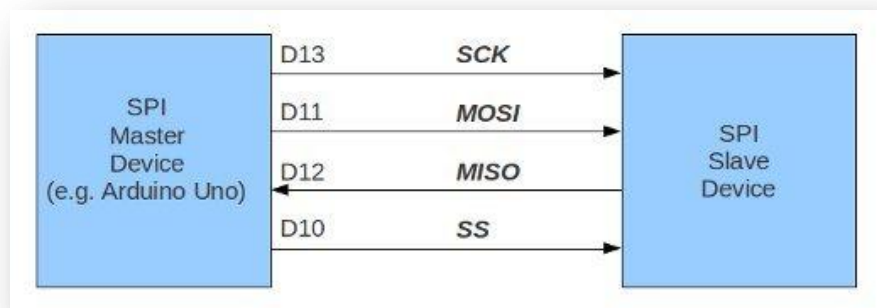


Figure 4.16: SPI Master Device with a single SPI Slave Device

Data travels back and forth along the MOSI and MISO lines between our Arduino and the SPI device. This can only happen when the SS line is set to LOW. In other words, to communicate with a particular SPI device on the bus, we set the SS line to that device to LOW, then communicate with it, then set the line back to HIGH. If we have two or more SPI devices on the bus, the wiring would resemble the following:

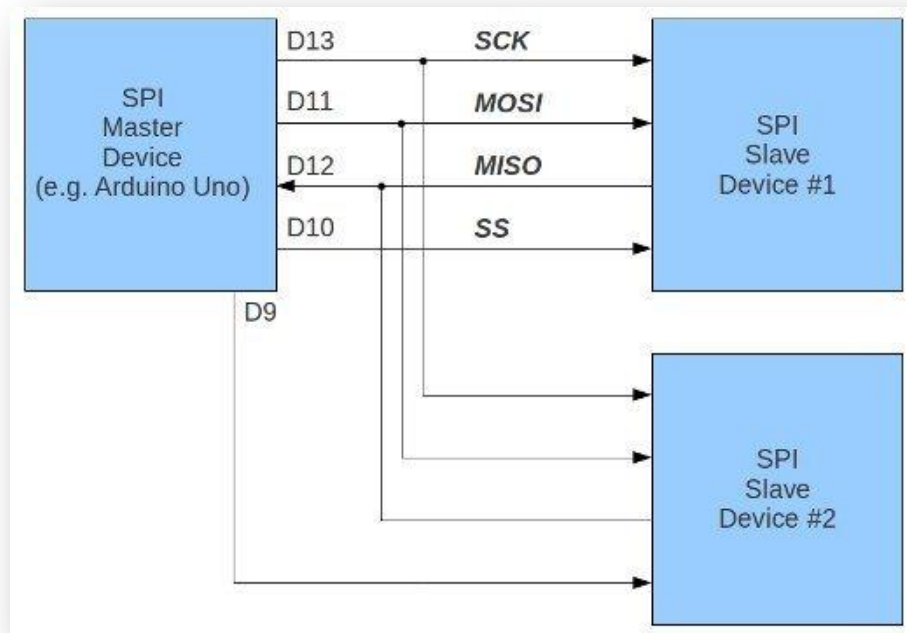


Figure 4.17: SPI Master is connected with multiple SPI slave devices

Notice how there is two SS lines – we need one for each SPI device on the bus. We can use any free digital output pin on Arduino as an SS line.

Data is sent to the SPI device in byte form. Eight bits make one byte, therefore representing a binary number with a value of between zero and 255. When communicating with our SPI devices, we need to know which way the device deals with the data – MSB or LSB first. MSB (most significant bit) is the left-hand side of the binary number, and LSB (least significant bit) is the right-hand side of the number.

Apart from sending numerical values along the SPI bus, binary numbers can also represent commands. Anyone can represent eight on/off settings using one byte of data, so a device's parameters can be set by sending a byte of data. These parameters will vary with each device and should be illustrated in the particular device's data sheet.

Table 4.1: Functions of 7 pins of NRF24L01+

Pin No. nRF24L01+	Name of the Pin	Function	Function during Transmission	Function during Receiving
1	GND	Ground		
2	VCC	Connect with 3.3V		
3	CE	Control data in Tx, Rx mode	Low-to-high transition while transmitting	High signal while receiving
4	CSN	Chip select not. Act as active low. When It is low it communicates with master. When it's high it ignores master		
5	SCLK	Serial Clock which is generated by Master and uses 8 bit shift resistor for a bite of data.	nRF24L01+ being as Slave read the state of the MOSI pin.	When clock pulses Master being Arduino board read the data from MISO pin.
6	MOSI	Master output slave input	Master transmit slave receive. While transmission arduino being a master, wireless module becomes slave. Master sends data bit where slave read the data. And thus the transceiver NRF24L01+ receives command from Arduino.	
7	MISO	Master input slave output		In this case Master will receive where slave will transmit. Here, Arduino becomes a master will only receive from slave alias the wireless module. The commands that will be received by NRF24L01+ will be sent serially through

				MISO line and Arduino will read from the same line.
--	--	--	--	---

4.11. Experimental Data analysis of nRF24L01+ with FireBOT:

A number of RF signals are observed in RIGOL DS1052E DS 1052E 2 Channel Digital Oscilloscope whilst experimenting in the midst of FireBOT and nRF24L01+ Transceiver wireless module.

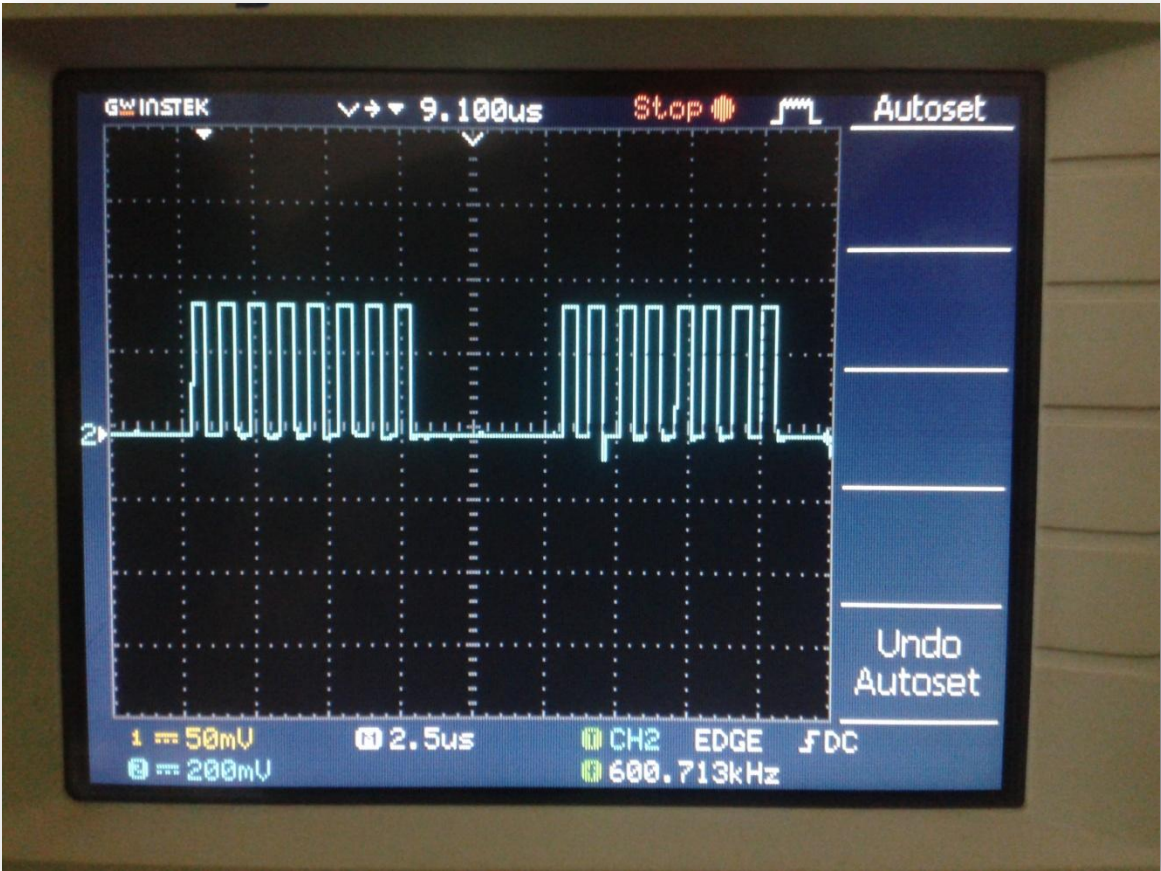


Figure 4.17: Digital Oscilloscope signal observation for transmission

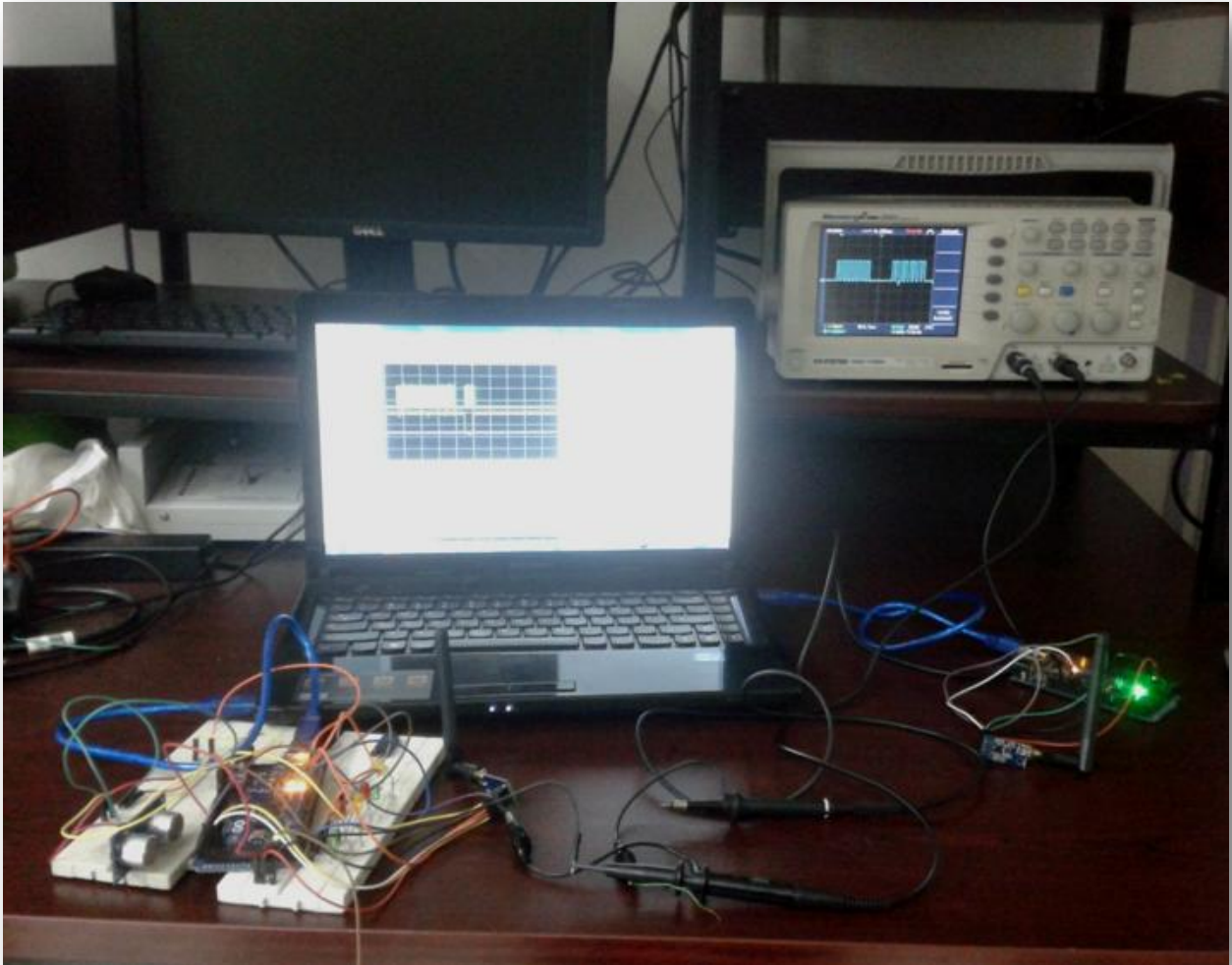


Figure 4.18: Laboratory setup and controlling RF Communication System

All the graphs in singular conditions are specified below by acquiring original .BMP format picture without altering it into SD Memory Card from Oscilloscope in laboratory.

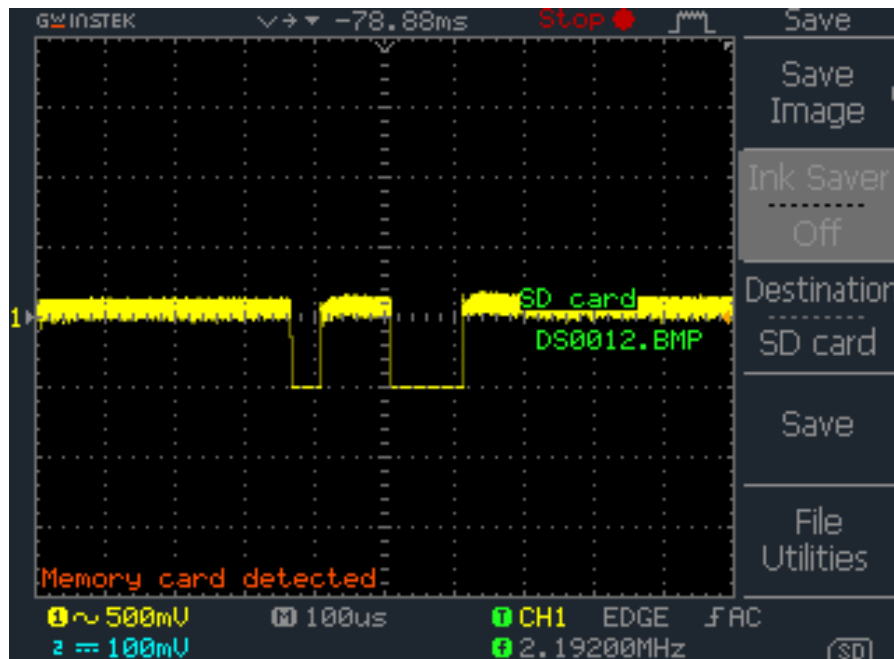


Figure 4.19: Signal at CE pin while Transmission

Figure 4.16 shows condition of signals at CE pin whilst nRF24L01+ is occupied for Transmission. CE is the Pin of Arduino module which controls transferring of data for Transmission and Reception modes. The above graph shows stipulation of signals at the instant when data were transmitted. Experimentally it is found that, the amplitude of square wave signal is about 500 mV (0.5V) and a moderately distorted signal is transmitted through air. Data are transmitted in the form of digital pulses which are completely observed in laboratory by the help of Digital Oscilloscope.

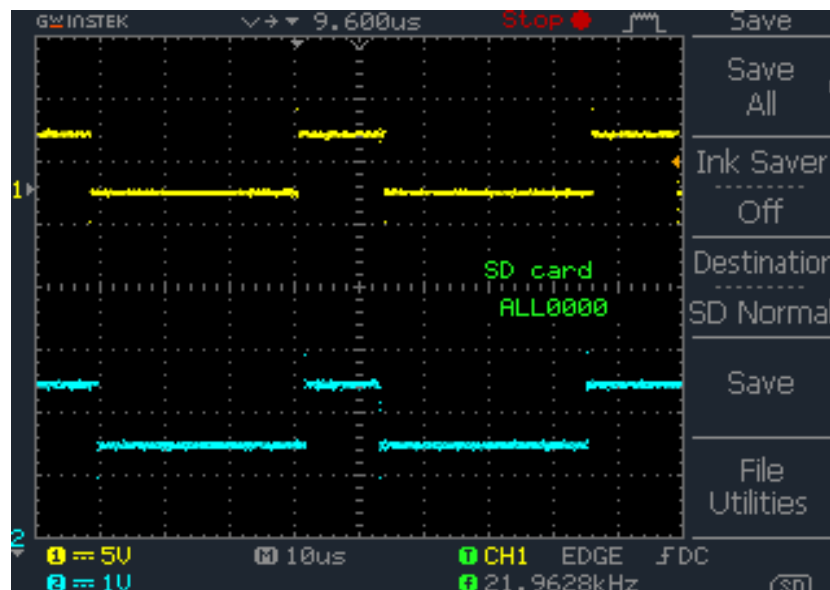


Figure 4.20: Signal at CSN pin while Transmission

Here, the Figure 4.20 shows Signal at CSN pin while Transmission is on process. The basic characteristic of CSN is due to Chip selection. By default it acts as active low which means when it gets a Logic Low ($\approx 0V$), it communicates with Master. On the other hand, when it gets Logic High ($\approx 5V$), it ignores the Master. The above statement is checked by laboratory experiment, though experiment shows a slight deviation from ideal state, but still it is worth of it.

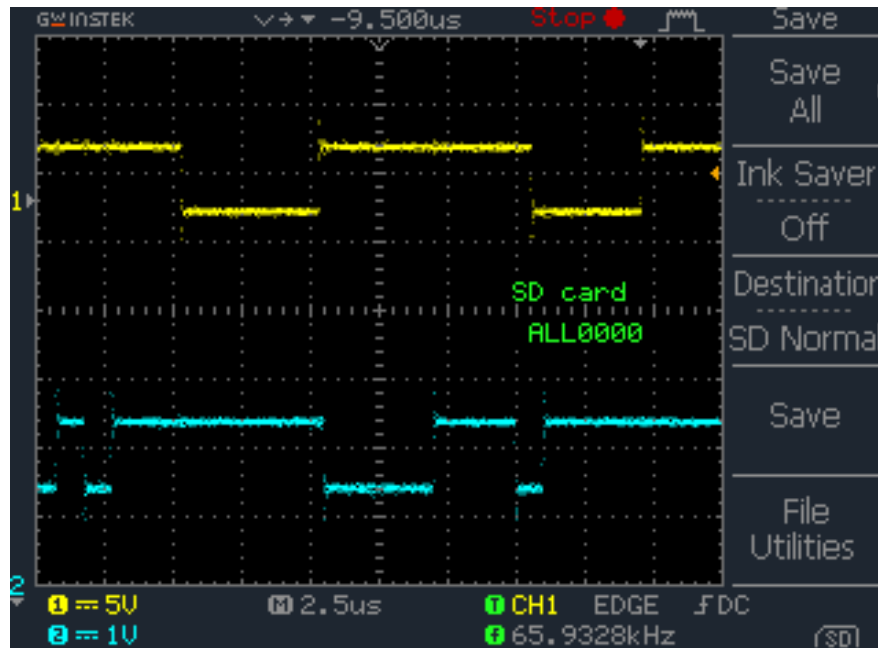


Figure 4.21: Signal at MOSI pin while Transmission

The condition of signal is depicted in the above Figure 4.21 when Arduino's MOSI pin is in action. MOSI refers to "Master Output Slave Input". The task of MOSI is to make Master Transmit and Slave Receive. Furthermore, while transmission is in process, Arduino (ATmega2560 Microcontroller) becomes the master and wireless module (nRF24L01+ Transceiver Module) becomes the slave. So, the Master sends data bit whereas slave reads the data and another Microcontroller based Arduino does work accordingly. In the above Digital Oscilloscope generated graph, it is observed that Yellow graph shows signal proviso at Arduino and Light Blue color graph shows condition of signal at slave that is on MOSI pin of nRF24L01+ module.

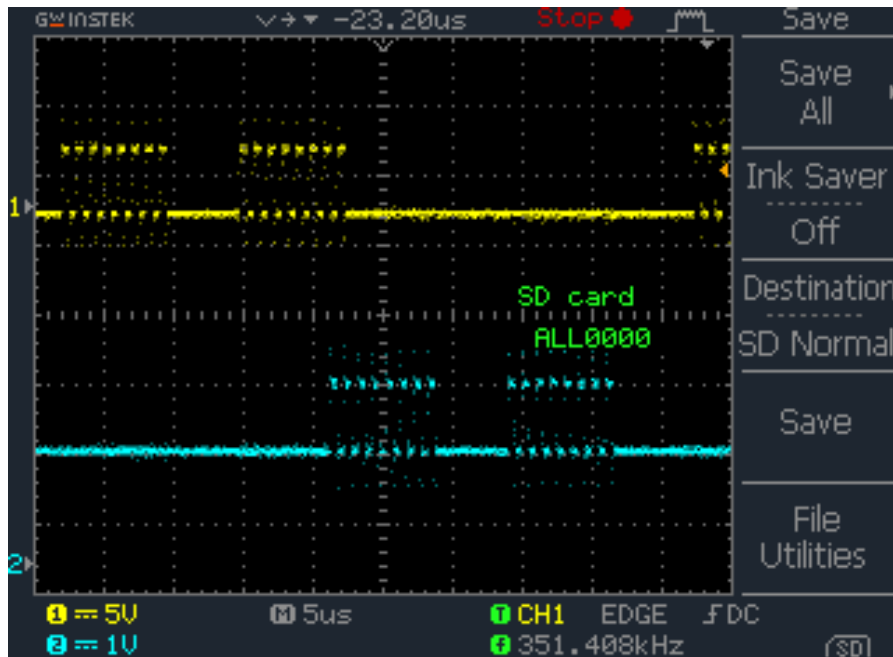


Figure 4.22: Perfect Signal at SCK pin while Transmission

Serial Clock which is generated by Master and uses 8 bit shift resistor for a bite of data. Here, nRF24L01+ wireless module being as Slave reads the state of the MOSI pin. On that time, clock pulses Master being Arduino board reads the data from MISO pin. Both of the signals are depicted in above graphs.

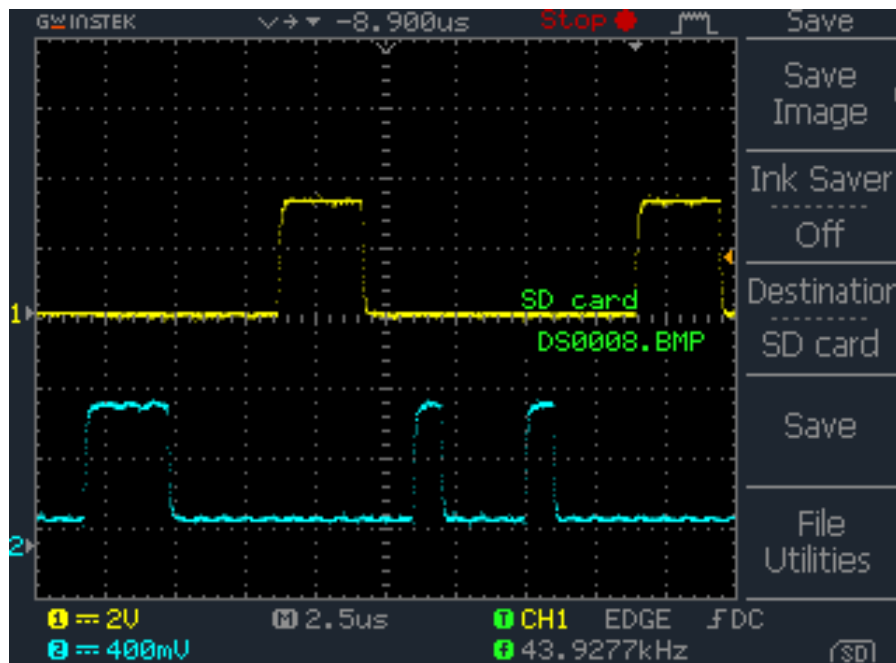


Figure 4.23: Signal at MISO pin while Transmission

MISO means Master Input Slave Output In this case Master will receive where slave will transmit. Here, Arduino becomes a master and it will only receive from slave alias the wireless module.

The commands that will be received by nRF24L01+ will be sent serially through MISO line and Arduino will read from the same line. The above graph Figure 4.23 shows validation of statements.

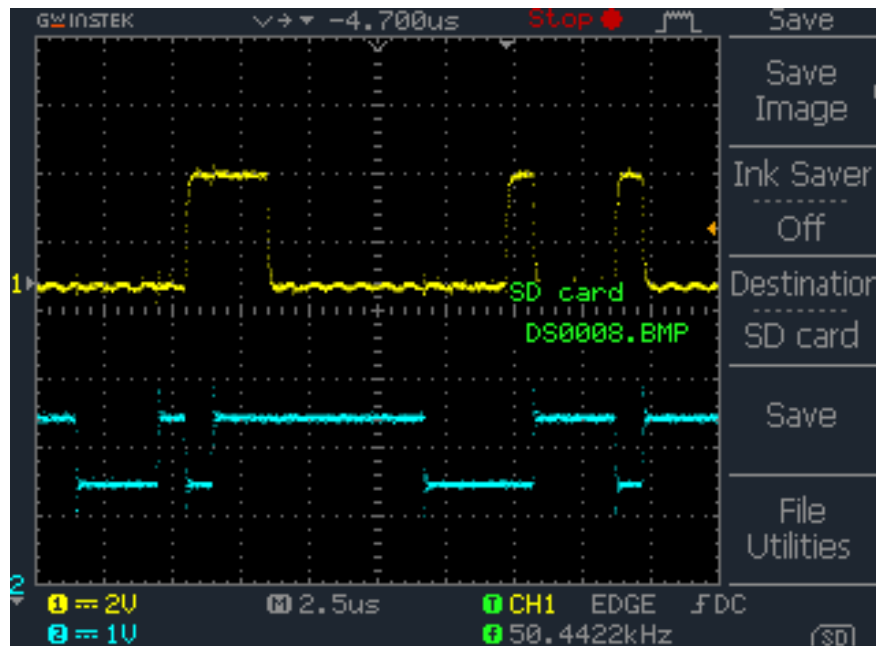


Figure 4.25: Signal at MISO (Yellow) and MOSI (Blue) in Reception Mode

In the above Figure 4.25, Yellow and Blue signals represent MISO and MOSI pins output respectively in Reception Mode. Theoretically, MOSI's signal will be inverted to MISO's signals and from oscilloscope generated graph it was strongly verified which indicates the successful reception and transmission of data through 2.4 GHz wireless signals.

Chapter -5

Fire Extinguishing System Design

5.1 Extinguishing Unit

5.1.1 Fire Extinguisher:

A fire extinguisher is an active device which is used to extinguish or to control small fires. A fire extinguisher consists of a hand-held cylindrical pressure vessel containing an agent which can be discharge to extinguish fire, pressure gauge, nozzle, safety pin, high pressure gas canister.

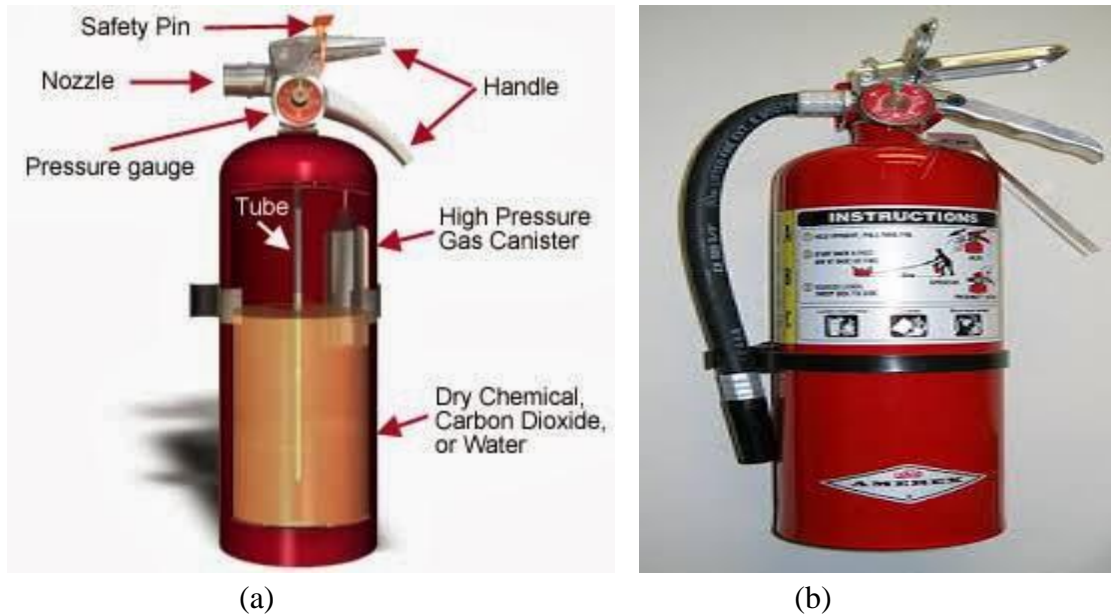


Figure 5.1: (a) Different Parts of a fire extinguisher; (b) A stored-pressure fire extinguisher [45]

Types of fire extinguishers:

There are main two types of fire extinguisher [44]

1. Stored pressure extinguisher: Expellant is stored in the same chamber as the firefighting agent itself. Stored pressured fire extinguishers are the most common type.
2. Cartridge-operated extinguisher: It contains the expellant gas in a separate cartridge that is punctured prior to discharge. This is not commonly used.

Fire extinguishers are further divided into handled and cart-mounted. Handheld extinguishers weight from 0.5 to 14 kilograms. Cart-mounted units typically weight more than 23 kilograms.

The commonest types of fire extinguisher are coded so that we can identify them.

Water extinguisher-red:

A red color coding indicates that the contents of the fire extinguisher are water. These are the cheapest and most widely used fire extinguisher. Used for class A fires. Not suitable for class B (Liquid) fires, or where electricity is involved.



Figure 5.2: water extinguisher

Dry powder extinguisher –Blue or Red with a Blue strip:

May be classified either ABC(containing ammonium phosphate) or BC (containing sodium or potassium phosphate), indicating the type of fire that can be tackled can be used on solids such as wood, plastics and liquids such as wood, plastics, and liquids such as petrol/ paints.



Figure 5.3: Dry powder extinguisher [2]

✚ Foam extinguisher- Cream or Red with a Cream strip:

Fire extinguisher coded with a cream color indicates that is a foam fire extinguisher. It can be used on solids such as wood, plastics and liquids such as petrol/ paints. It is Safe to use it near electrical appliances. It is extinguished by smothering the flames.



Figure 5.4: Foam extinguisher [46]

✚ CO₂ extinguisher- Black or Red with a Black strip:

Extinguishers that come with a black or red strip coding indicate that it is a carbon dioxide or CO₂, extinguisher. This is the recommended method of fighting class E electrical fires and it is also highly effective against classes B liquid fires. The thing to remember about this type of fire extinguisher is that it provides no security for after the fire is out, meaning that the fire could start again. It is safe to use near electrical appliances and extinguished by displacing oxygen. In addition, it is not suit for solids [45].



Figure 5.5: CO₂ extinguisher

Table 5.1: Types of extinguisher, agent, and class and sample application

Extinguisher type	Agent	class	Sample applications
Multi-purpose Dry Chemical	Mono-Ammonium phosphate	ABC	Office, Hotel Schools and Warehouses
Regular Dry Chemical	Sodium Bicarbonate	BC	Vehicles, Training and Laboratories
Purple K Dry Chemical	Potassium Bicarbonate	BC	Oil Industry , Airport Ramps, Military and Fuel Service
CO₂	Carbon Dioxide	BC	Factories and Processing Plants
Water	H ₂ O	A	Storerooms, Barns and Attics
Foam	AFFF/FFFP	AB	Fueling areas, Manufacturing and Construction

The type of fire extinguisher used in our thesis experiment and why?

We have used two dry powder fire extinguishers. Each extinguisher weight is 6kg. It contains ammonium phosphate. The logic behind using dry powder extinguishers because these are especially effective against class B fires, but can be useful to help exterminates fires that fall under classes A and C as well. It is for this reason a dry powder fire extinguisher is known as a “multi-purpose” extinguisher. Besides, it can be an effective tool to help put out fires resulting from gas. In addition others features of dry powder are-

1. Squeeze grip operation
2. Corrosion resistant finish
3. Protective plastic base

Sprayer operation:

Before spraying operation handle are punched manually. The solenoid valve is connected to the nozzle. Nozzle is control by solenoid valve. When robot reaches the firing location that time solenoid valve is activated and nozzle release the extinguishing agent at a specific rate and attenn to quickly extinguish a fire. After extinguishing fire, nozzle is turned off by solenoid valve.

5.1.2 Solenoid valve

What is Solenoid Valve?

Solenoid valves are electrically actuated mechanical systems that control the flow of substances like liquids and gases. Solenoid valves are commonly used on automotive transmissions. They allow an automatic transmission to shift gears through controlled fluid flow.

5.1.2.1 Mechanism of Solenoid valve:

Solenoid valves are also characterized by how they operate. A small solenoid can generate a limited force. If that force is sufficient to open and close the valve, then a direct acting solenoid valve is possible. An approximate relationship between the required solenoid force F_s , the fluid pressure P , and the orifice area A for a direct acting solenoid valve is [47]:

$$F_s = PA = \frac{P\pi d^2}{4}$$

Where d is the orifice diameter

A solenoid valve has two main parts: the solenoid and the valve. The solenoid converts electrical energy into mechanical energy which, in turn, opens or closes the valve mechanically. A direct acting valve has only a small flow circuit, shown within section E of this diagram (this section is mentioned below as a pilot valve). In this example, a diaphragm piloted valve multiplies this small pilot flow, by using it to control the flow through a much larger orifice.

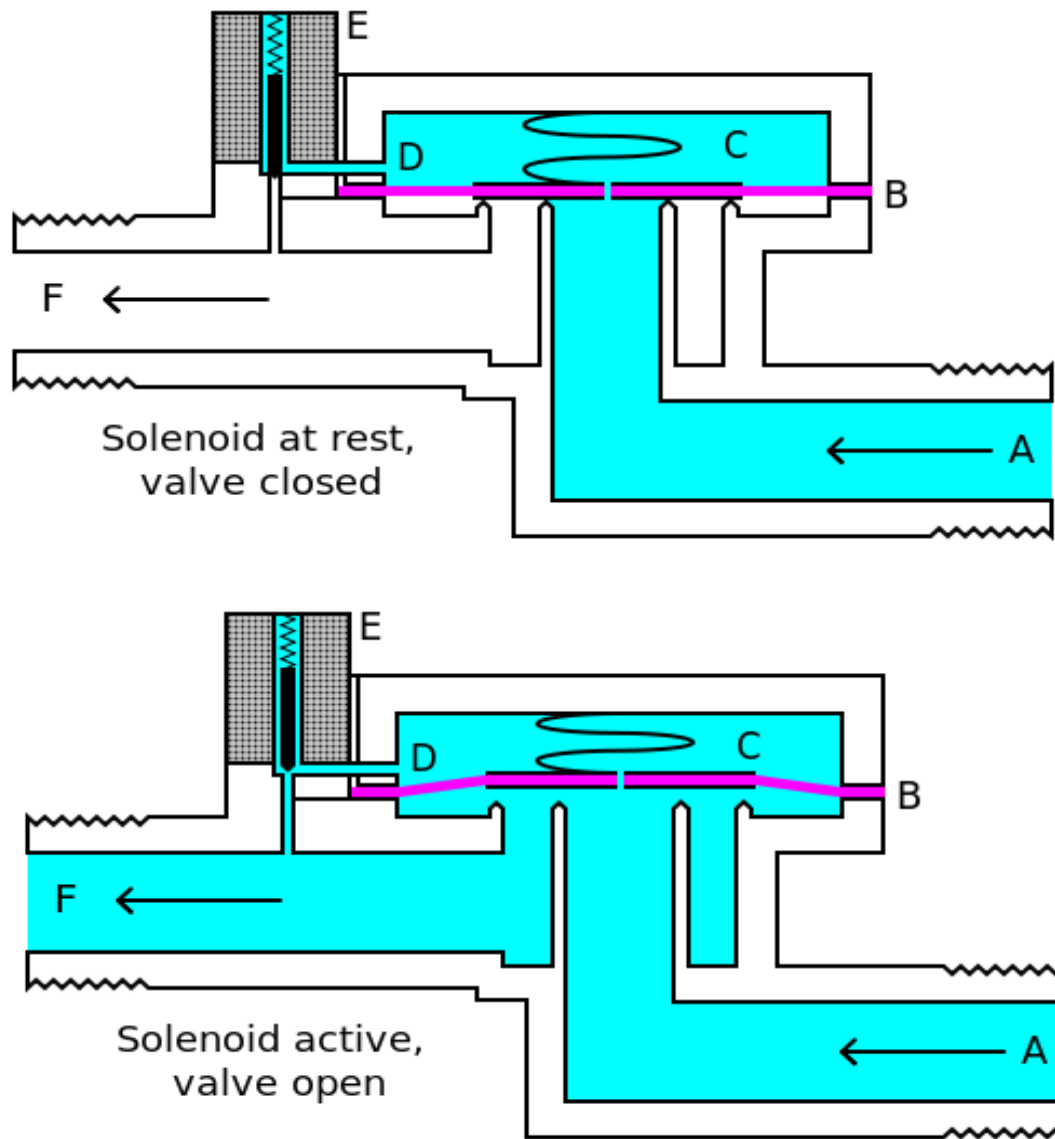


Figure 5.6: Design of a basic solenoid valve [48]

A-Input side

B-Diaphragm

C-Pressure Chamber

D-Pressure Relief

E-Solenoid

F-Output side

The diagram to the right shows the design of a basic valve, controlling the flow of dry gas in this example. At the top figure is the valve in its closed state. The gas under pressure enters at A. B is an elastic diaphragm and above it is a weak spring pushing it down. The diaphragm has a pinhole through its center which allows a very small amount of gas to flow through it. This gas fills the cavity C on the other side of the diaphragm so that pressure is equal on both sides of the diaphragm; however the compressed spring supplies a net downward force. The spring is weak and is only able to close the inlet because water gas is equalized on both sides of the diaphragm. Once the diaphragm closes the valve, the pressure on the outlet side of its bottom is reduced, and the greater pressure above holds it even more firmly closed. Thus, the spring is irrelevant to holding the valve closed.

The above all works because the small drain passage D was blocked by a pin which is the armature of the solenoid E and which is pushed down by a spring. If current is passed through the solenoid, the pin is withdrawn via magnetic force, and the gas in chamber C drains out the passage D faster than the pinhole can refill it. The pressure in chamber C drops and the incoming pressure lift the diaphragm, thus opening the main valve. Gas now flows directly from A to F.

When the solenoid is again deactivated and the passage D is closed again, the spring needs very little force to push the diaphragm down again and the main valve closes. In practice there is often no separate spring; the elastomer diaphragm is molded so that it functions as its own spring, preferring to be in the closed shape.

The types of valve we are using:

The most common solenoid valve is the two way valve. A two way valve only has two ports, whereas more advanced designs may have three or more, depending on what it will be used for. In our thesis research two way valves is used for controlling the extinguisher material. On a two way valve each of the two ports is alternately used to permit flow as well as close it off.

5.1.2.2 Use of Solenoid valve in Extinguisher

A two way valve can be specified to be either “Normally Open” or “Normally Closed” in its operation. With a normally open valve, the valve remains open until some type of current is applied to close the valve. Suspension of the electrical power causes the valve to automatically re-open to its normal state. A normally closed solenoid valve is the most common, working in the opposite fashion, remaining closed until a power source causes it to open.

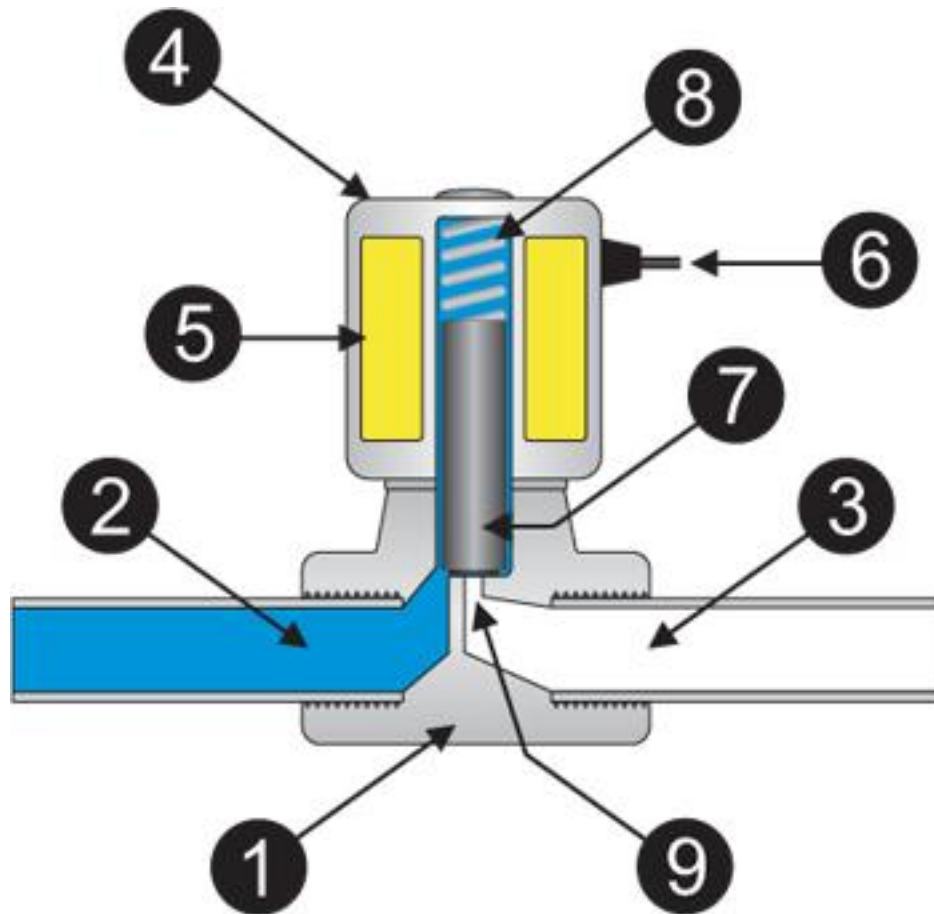


Figure 5.7: Different parts of a Solenoid Valve

- | | | |
|----------------|------------------|------------|
| 1. Valve Body | 4. Coil/Solenoid | 7.Plunger |
| 2. Inlet Port | 5 .Coil Windings | 8. Springs |
| 3. Outlet Port | 6. Lead wires | 9.Orifice |

The media controlled by the solenoid valve enters the valve through the inlet port. The media must flow through the orifice before continuing into the out port. The orifice is closed and opened by the plunger.

5.1.2.3 Control Circuit of Solenoid Valve:

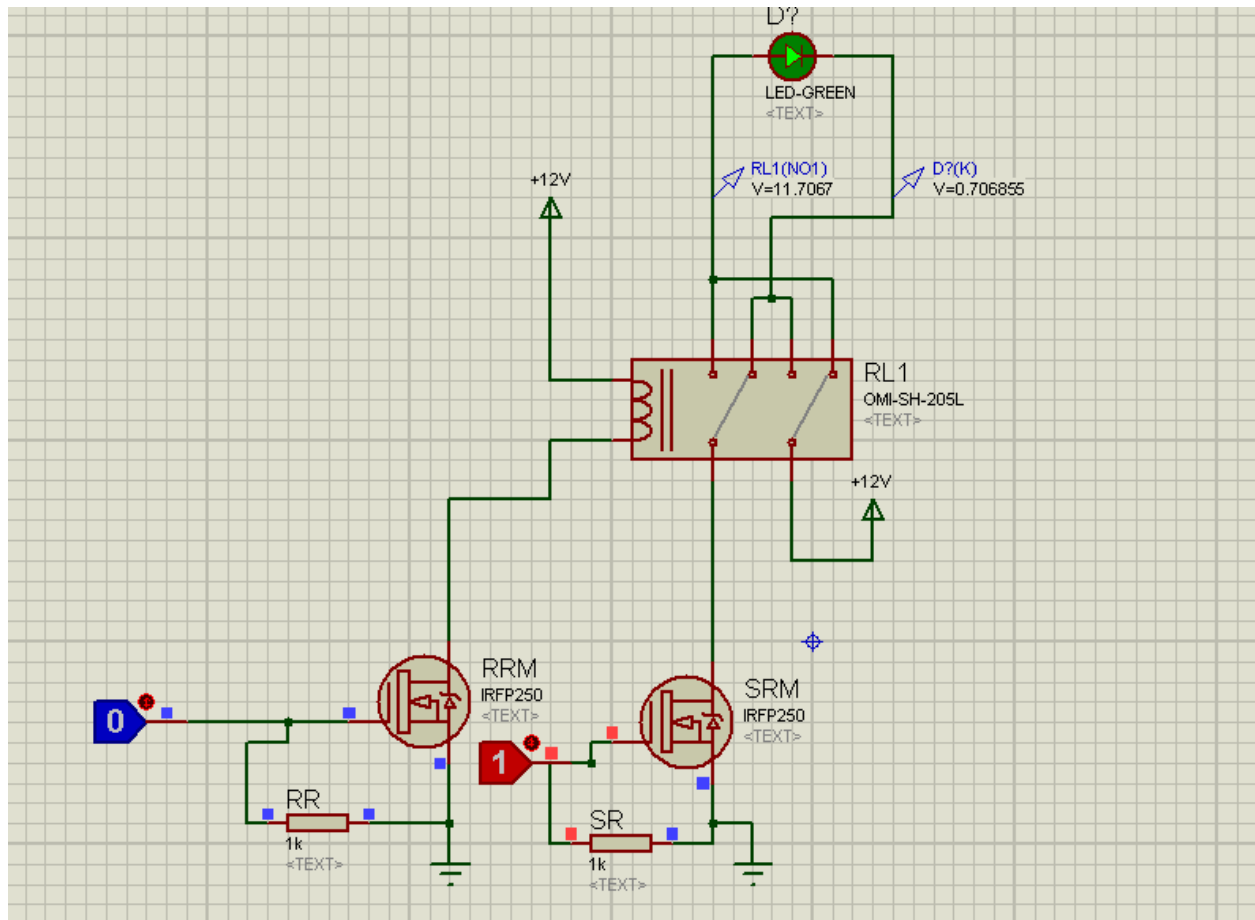


Figure 5.8: Solenoid Valve Control Circuit Simulation (Solenoid valve turned on)

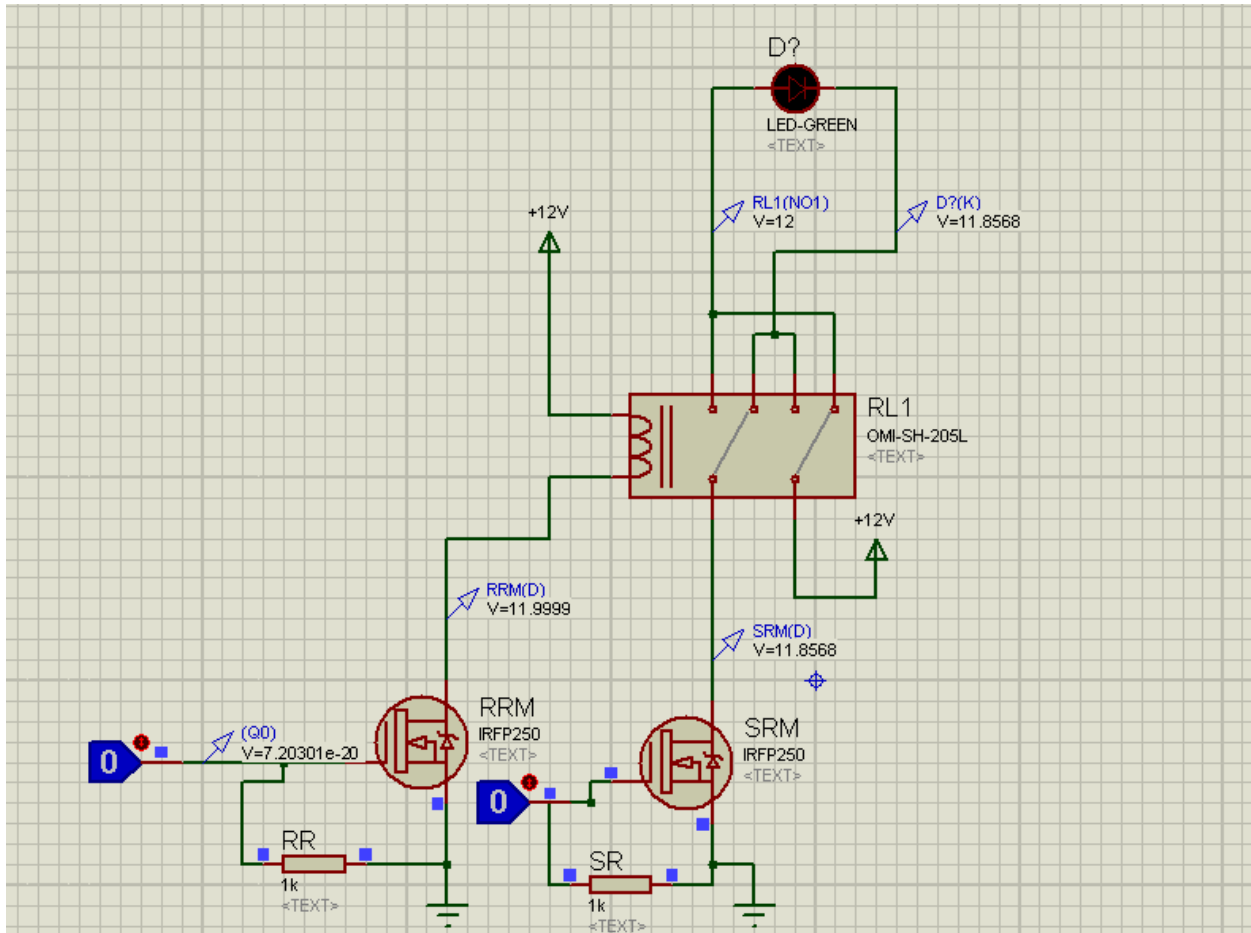


Figure 5.9: Solenoid Valve Control Circuit Simulation (Solenoid valve turned off)

In this research work, solenoid valve is used to manage the extinguisher remotely from the base station. In this controlling circuit, a dual pole solenoid valve, 2 N-channel MOSFET is used to manage high voltage and current in case applied.

From the base station, corresponding keyboard input will be pressed to wirelessly transfer the command to robot's Arduino board which will be in charge of turning the solenoid valve system to ON or OFF.

The signal from the Arduino Mega2550 digital I/O pins goes to the gate of the IRFP250N Mosfets that are responsible of acting as logic gates. Because N-channel MOSFET operates in positive high input voltage with high input resistance, whenever it gets 5V the resistance lowers to around 10KΩ that drain and source terminal literary shorted which act as switch ON.

In this solenoid valve actuator control circuit two n-channel MOSFETs are used. One MOSFET denoted as SRM act as switch, if it turned on battery power connects to the solenoid part and the actuator powered up with sufficient voltage to turn up.

5.1.3 Rotation of Extinguisher nozzle:

To make the fire extinguishing robot more efficient we have included a rotational system to turn around the nozzle of extinguisher according to the wireless command.

When, the robot gets a command “P” from its operator, it will start rotating clockwise for 10 degrees. The system is design in such a way that, it can +/- 10 degrees of rotation. That means wireless command of “O” will rotate the nozzle anti-clockwise. One thing to be noticed that, servo motor should not rotate more than 80 degrees because of nozzle line feasibility.

Chapter -6

Sensor System Design

6.1 IP Camera:

IP is the Internet Protocol. An IP camera is a networked digital video camera that transmits data over a Fast Ethernet link. IP cameras (also called "network cameras") are most often used for IP surveillance, a digitized and networked version of closed-circuit television (CCTV).

An IP camera usually consists of various components, including a lens, an image sensor, and one or more processors. Memory plays the role of a hard drive by storing video content and the firmware needed to operate the device. Processor is required to process images, compress video, and perform a wide range of network function.

Types of IP cameras:

IP camera can be classified on the following basis:

- A. Camera lens and image processing unit constructed-
 1. Integrated IP Camera: in which the image capture unit, like camera lens, is integrated with image processing unit.
 2. Video server: In which only has image processing unit and other units such as image transfer, networking module. There is no video capture unit inside it.
 - B. Video capture unit connected to video server-
 1. Corded camera: In which the camera is connected by cord.
 2. Cordless camera: In which the camera is connected by radio signal or X10 adaptors.
 - C. IP camera connected to internet-
 1. Wired camera: In which the camera is connected with a CAT 5 standard network cable through a RJ 45 jacket of Ethernet.
 2. Wireless camera: In which the camera is connected to internet through WiFi, Bluetooth or other connecting method. We are using this type of camera in our thesis.
- [49]

Features:

It has many features which are required to complete our experiment like:

- IP cameras are web-ready which can provide surveillance from anywhere there is an internet connection.
- Digital zoom.
- The ability to easily send images and video anywhere with an Internet connection.
- Progressive scanning, which enables better quality images extracted from the video, especially for moving targets
- Adjustable frame rates and resolution to meet specific needs.
- Two-way communication.

- The ability to send alerts if suspicious activity is detected.
- Lower cabling requirements.
- Support for intelligent video
- Since we are considering to wireless communicate regarding our robot and we know that IP comers are also to function on a wireless network.



Figure 6.1: Wireless day/ night surveillance camera TL-SC3171G

TL-SC3171G uses 12 IR (Infrared) LEDs light as a means to produce clear images in the dark and an industrial mechanical ICR (IR Cut-filter Removable) completely filters that light during the day ensuring vivid video. Furthermore, using this IP camera in our robot body we will be able to decrease the property damages from fire by watching video and taking immediate steps.

6.1.1 Specifications [50]:

Table 6.1: Specifications of TL-SC3171G IP Camera

Specification		
Camera	Image Sensor	1/4" Progressive scan CMOS sensor
	Lens	F: 2.0, f: 4.3mm
	Viewing Angle	Diagonal 61°, Horizontal 48°Vertical 36°
	Digital Zoom	10x Digital
	Day/Night Mechanism	Mechanical IR-Cut filter Removable; 12 IR (Infrared) LEDs
	IR Working Distance	10 meters (33 feet)
Wireless	Wireless Data Rates	IEEE 802.11 b/g , Up to 54Mbps
	Frequency	2.4 – 2.4835 GHz
	Wireless Transmit Power	<20dBm(EIRP),
Interfaces	Network Interface	1 RJ- 45 for Ethernet 10/100 Base-T
	Power Connector	DC power jack
	Button	Reset push button
	External Power Supply	12V DC, MAX 12W
System Requirements	CPU	Pentium4 1.8GHz
	Graphic Card	512MB RAM
	Supported Browser	Internet Explorer; Firefox; Chrome; Safari

6.1.2 Communication Techniques with Fire Fighting Robot:

IP camera can send and receive data via a computer network and the internet. In our thesis work, IP camera is set on our robot body and it is connected to an IP network such as a LAN (Local Area Network) or internet. Through the network, the camera sends stream video which is accessed remotely our monitor computer on the network. The IP camera transmits video bit streams based on HTTP/TCP (Hypertext Transport Protocol/Transmission Control Protocol) through the Internet. The image captured by the camera are digitized and compressed by the monitoring computer. Through the video streaming is also possible to identify any person existing on firing place. Camera will be monitored at all times changing camera position.

6.2 Interface with sensors

6.2.1 Temperature Sensor:

All physical parameters are temperature-dependent since the process control of chemical reaction is of major importance in temperature control. In view of the fact that temperature is undoubtedly the most widely measured variable that the accuracy cannot be overstated. From the household kitchen's refrigerator to chemical reactor monitoring of temperature sets possible maintenance and control in order to function properly. Besides for accurate temperature control precise measurement of temperature required thus there stands the selection of perfect sensor for temperature measurement. Although there are different types of temperature sensors, each has their own characteristics in terms of their applications. Therefore, while choosing sensor several factors must be considered: temperature range, response time, accuracy, linearity and sensitivity [51].

Temperature sensing can be done either in direct contact with the heating source or remotely, indirect contact. Temperature sensors come in a wide variety including thermocouples, resistance temperature detectors (RTDs), thermistors, semiconductor sensors: analog output sensors, digital output sensors.

However, sensor and IC manufacturers offer four types of temperature sensors:

1. Analog output temperature sensors:

Thermistors and silicon temperature sensors show analog output in results. However, in terms of

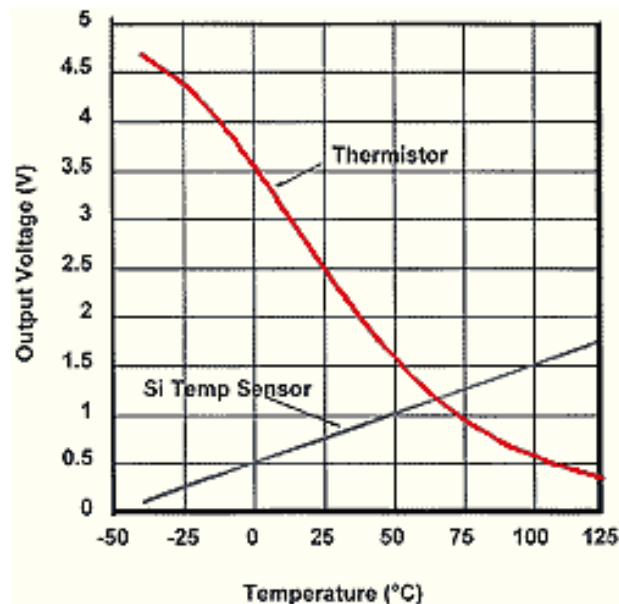


Figure 6.2: Linearity of thermistor and temperature sensor [52]

linearity between voltage and temperature silicon based sensors have better linear relationship than thermistor [52].

2. Digital I/O sensors:

Digital temperature sensors eliminate extra components like A/D converter, amplifier as it has precision output in degree Celsius. These sensors have I2C or SPI interface with microcontroller that reports temperature readings in microcontroller. However, there is no linearity between temperature and output voltage.

3. Analog plus sensors:

Analog plus sensors are generally used in measurement applications. The logic output trips when a specified temperature rises above a fixed threshold. Here, frequency is proportional to temperature thus microcontrollers counter measure time and calculate the ambient temperature [52].

4. System monitor IC:

System monitor IC provides digital output therefore monitor system supply voltages. When voltage surpasses the limit provides alarm.

Among all of these sensors, in our research work we have decided to employ analog output sensor because of its linearity between temperature and output voltage. In this case, we have used a precision integrated-circuit temperature sensor, LM35.

6.2.1 1 General Description-LM35:

The Lm35 precision integrated-circuit temperature sensor converts temperature to voltage through a temperature dependent resistor. Since, output voltage is linearly proportional to temperature (centigrade), user do not need to subtract large constant voltage for scaling. The device uses single power supplies and draws 60 μ A from the supply and it has a low self heating of <0.1° C in still air. The Lm35's low output impedance, linear output and precision inherent calibration make interfacing to control easily. The output voltage is 10mV/degree centigrade when the supply voltage is 5V, so if output is 3000 mV then the temperature is 30 degrees.

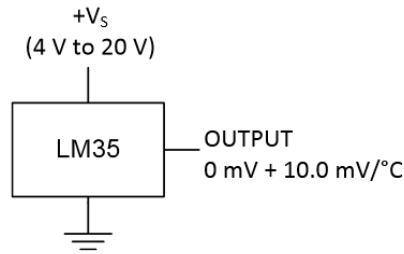


Figure 1. Basic Centigrade Temperature Sensor (+2°C to +150°C)

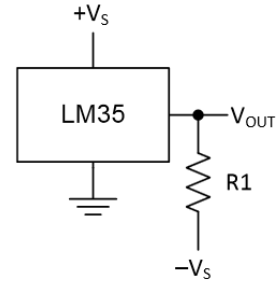


Figure 2. Full-Range Centigrade Temperature Sensor

Figure 6.3: Circuit connection of LM35

Above there are two types of circuits shown to apply, however we used basic centigrade temperature sensor since the robot will be in hot environment, and there is no chance to fall temperature in minus degree.

Features [53]:

- Calibrated directly in ° Celsius
- linear 10mV/° C
- Accuracy of 0.5° C
- Rated for full -55° C to 150° C range
- Suitable for remote application
- Low cost due to wafer-level trimming
- Operates from 4 to 30 V
- less than 60μA current drain
- Low self-heating
- Low impedance output, 0.1Ω for 1 mA load

Pin diagram:

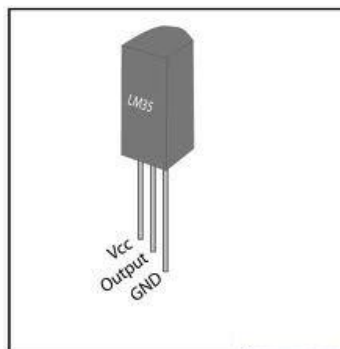


Figure 6.4: Pin diagram of LM35

Table 6.2: Description of LM35 Pin

Pin No.	Name	Function
1	VCC	Supply Voltage (-0.2V to 35V)
2	Output	Output voltage (1V to 6V)
3	GND	Ground

6.2.1.2 Temperature Measurement:

The output pin of LM35 is connected with the pin of A0 of Arduino Mega 2560 where pin A0-A15 are analog pin of arduino mega 2560. Since the temperature sensor that is being used in this research is analog-output, analog pin of arduino mega 2560 is connected with LM35 to read the ambient temperature.

The Atmega that used in arduino mega 2560 has onboard 16 channel analog-to-digital (A/D) converter. The converter supports 10 bit returning integers from 0 to 1023. Since, reference voltage is 5V preset for this study, digital value for 5V is 1023.

From the ADC pin, digital value can be read, so a calculation has been done to get the actual temperature from the digital value.

$$\text{digital value, 1023} = 5\text{V}$$

$$\text{digital value, x} = \frac{5x}{1023} \text{ V} = 0.48x \text{ mV}$$

Given that, output voltage of the sensor is linear with temperature,

$$10\text{mV} = 1^{\circ} \text{ C}$$

$$y \text{ mV} = \frac{y}{10}^{\circ} \text{ C}$$

6.2.2 Ultrasonic sensor:

Distance measurement of a moving entity is required in a large number of devices. Ultrasound sensors are very adaptable in distance measurement. They are useful for both the air and underwater [55].

Pin diagram:

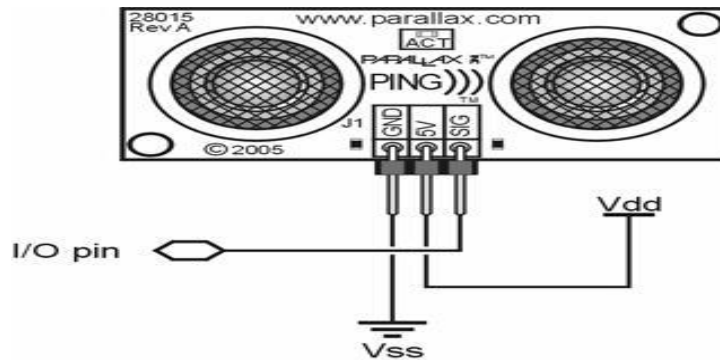


Figure 6.5: Pin diagram of TS-601

Table 6.3: Pin description of TS-601

Name of Pin	Function
GND	Ground
5V	Supply voltage 5V
SIG	I/O pin connected with microcontroller

Features:

- Supply Voltage: 5V DC
- Supply Current: <15mA
- Range: 2 cm to 3m
- Bidirectional TTL pulse
- Continual response time: 15 ms
- Impulse width output

6.2.2.1 Theory of operation:

Ultrasonic sensors emit short, high-frequency sound pulses (burst) above the human hearing range under control of a host microcontroller at regular intervals that propagate in the air at the velocity of sound. The emitted burst ranges to short 40 KHz. If they strike an object, then they are reflected back as echo signals to the sensor and the host microcontroller listen the echo. The burst travels through the air about 1130 feet per second. Internal timer then activated to measure the time duration of the whole travel. As the distance to an object is determined by measuring the time of flight and not by the intensity of the sound, ultrasonic sensor is excellent at suppressing background interference.

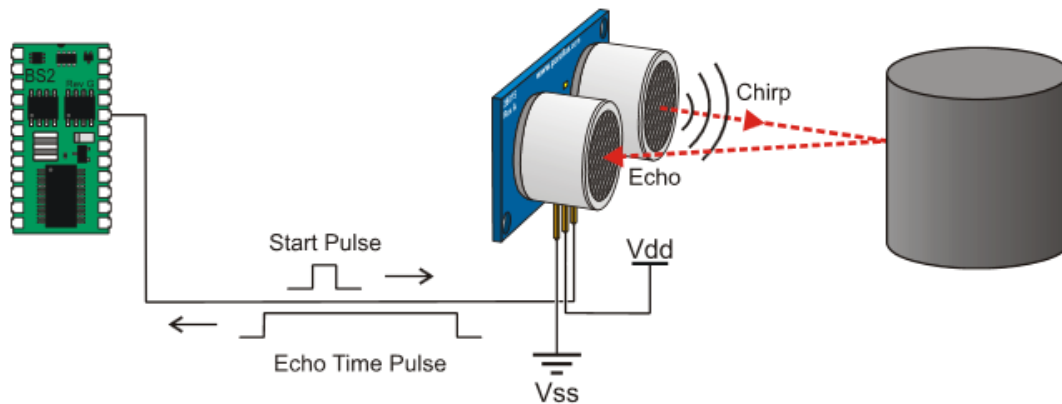


Figure 6.6: Distance measurement with the help of microcontroller

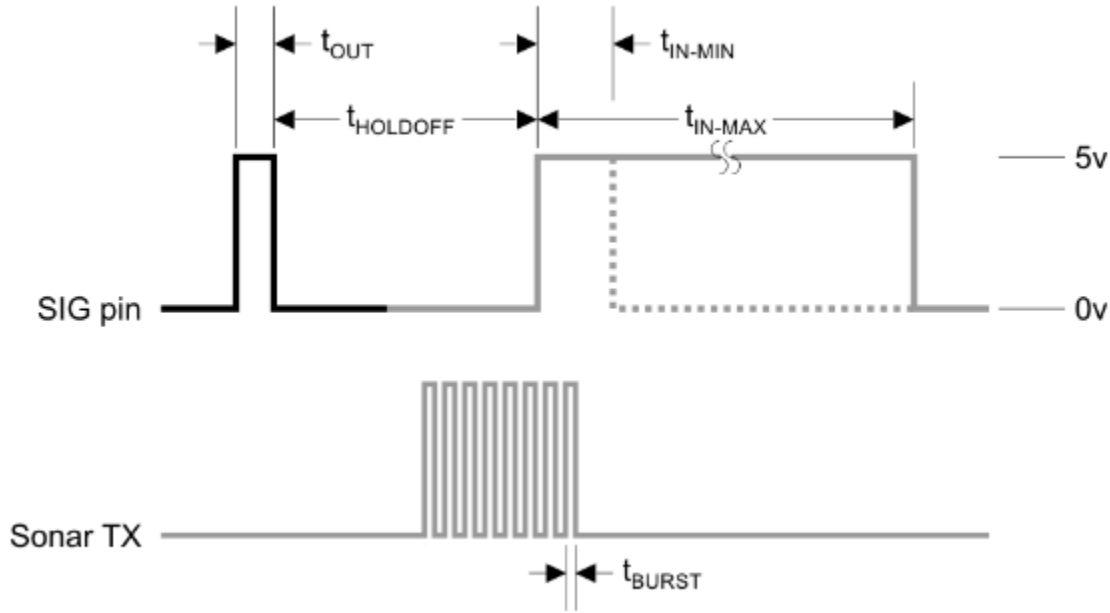


Figure 6.7: Theoretical pulses emission and receives through ultrasonic sensor

According to the figure [54],

$$t_{out} = > 5\mu s$$

$$t_{HOLDOFF} = 200 \mu s$$

$$t_{IN-MAX} = 200 \mu s$$

$$t_{BURST} = 25 \mu s$$

6.2.2.2 Distance Measurement:

Atmega2560 microcontroller based arduino mega 2560 has 12 PWM pins which are used to send or receive pulses and counts if necessary. Therefore arduino will transmit a pulse of 500 μs that will go to the sensor which will emit signal for that time frame and then receive echo. The onboard timer routine of arduino mega 2560 helps to get the time duration of echo receives. The **PulseIn()** function of arduino start calculating time whenever it gets HIGH till it goes to LOW and returns the length of pulse in microsecond.

We know, in constant velocity the equation of motion is, $S=VT$

Where, S = distance, V =constant velocity, T = time duration

However, the **PulseIn()** function give the distance aka length in terms of time, a conversion in coding has been done to get final output in inches. According to parallax's datasheet, sound travels at 1130 feet per second thus 73.746 microseconds per inch.

6.2.2.3 Experimental Measurement:

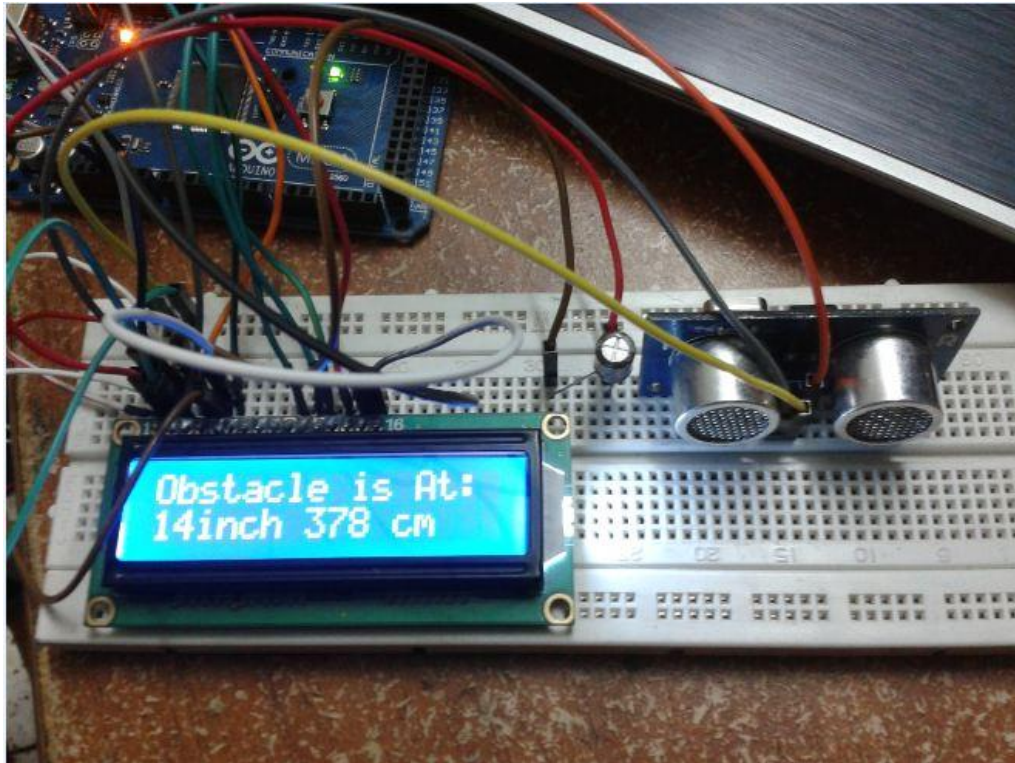


Figure 6.8: Circuit connection for Sonar Sensor

At simulation lab, using digital oscilloscope the waveforms on the SIG pin is measured, where the distance is 47 inch.

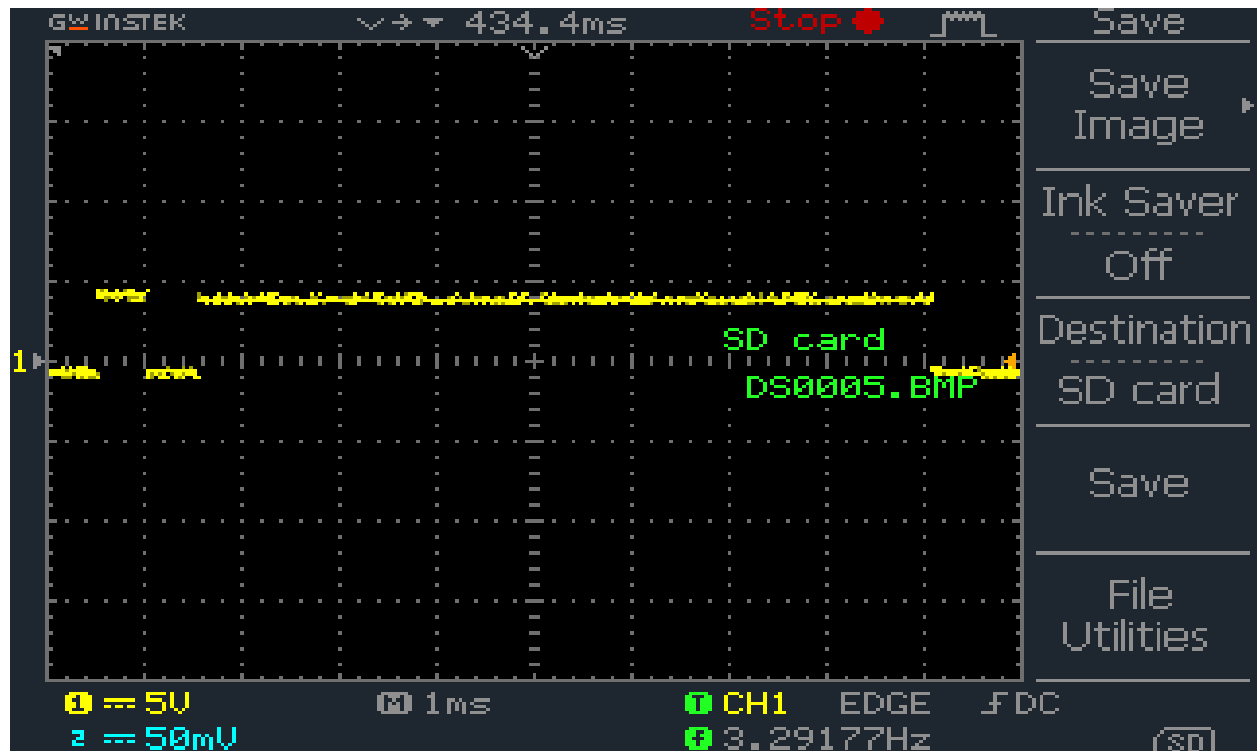


Figure 6.9: Oscilloscope observation of sonar sensor

From the above graph, first of all a pulse is seen which is about 500 μs in width and the returned pulse is 7500 μs long which is the duration of ultrasonic sounds return trip.

However, the predefined routine for detecting obstacle from the surrounding is by triggering the sensor in first place.

Pulse width sent from Arduino = 500 μs

From the graph, sensor starts receiving echo after 400 μs

Pulse width of received echo signal = 7500 μs

We know, $distance = \frac{speed\ of\ sound \times travel\ time\ of\ echo}{2}$

Also, it is known that sound travels through air at about 340 m/s or 74 inch/s

$$s(meter) = \frac{340 \times 7500\mu s}{2} = 1.27\ meter$$

We know, 1 meter = 39 inch

So distance of 1.27 in inches is 50 inch

Table 6.4: Experimental result from TS-601 sensor

Serial No.	Actual Distance (inch)	Travel Time	Measured Distance (inch)	% Error
1	3	420	4.2	40 %
2	5	740	6.5	30 %
3	8	1190	9.3	16.25 %
4	11	1640	11.5	4.54 %
5	13	1400	14.1	8.46 %
6	17	2500	19.1	12.35 %
7	21	2980	23.5	11.9 %
8	26	3800	27.4	5.38 %
9	30	4400	31.9	6.33 %
10	32	4700	34.9	9.06 %
11	40	6000	42.6	6.5 %
12	50	7500	51.45	2.9 %

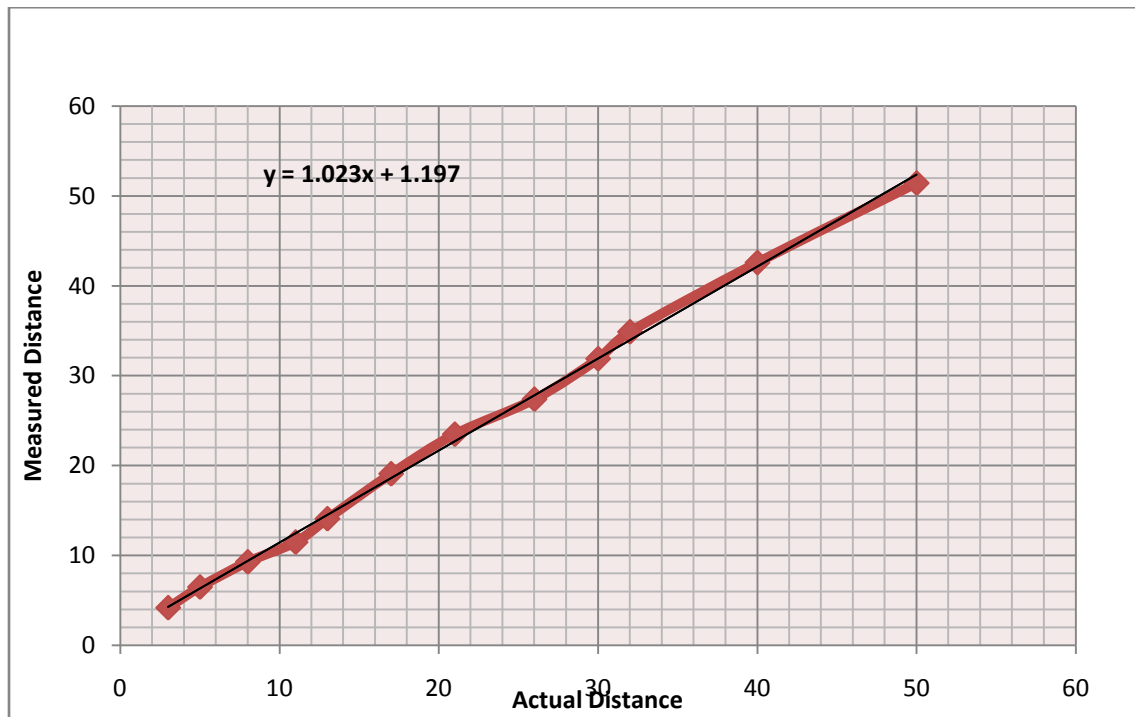


Figure 6.10: Graph between actual distance and measured distance

According to figure, it is seen that difference between actual and measured distance is contracted and short distances have most erroneous measurements than from the long range.

From the plotted graph, an equation has been formed, $y = 1.023x + 1.197$ which is a linear equation.

6.2.3 PIR sensor:

Passive Infrared (PIR) sensors are also known Pyroelectric Infrared sensors are ideal sensors because while they operate, their presence cannot be detected as in the active sensor cases. They detect difference in temperature, thermal radiation, in the environment caused by human body or an animal. PIR sensors are mostly used in integrated circuits where it take infrared signal as the input signal and convert it to analog electrical output signal.



Figure 6.11: Parrallax PIR sensor

Features [56]:

- TTL output
- High Sensitivity
- Delay Time: 5s to 18 minute
- Supply voltage: 5V to 20V DC
- Source current: 12mA to 23mA
- Range: 15ft to 30ft
- Operation temperature: -15 degree C to 70 degree C
- Infrared sensor
- Light sensor: CdS photocell

Pin Diagram:

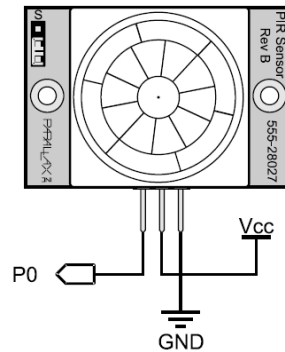


Figure 6.12: Pin diagram of Parallax PIR sensor

Table 6.5: Pin description of PIR sensor

Pin No.	Pin Name	Function
1	GND	Ground
2	VCC	Supply voltage: 3 to 6V
3	I/O pin	Signal pin; HIGH=movement, LOW=no movement

6.2.3.1 Working Principle:

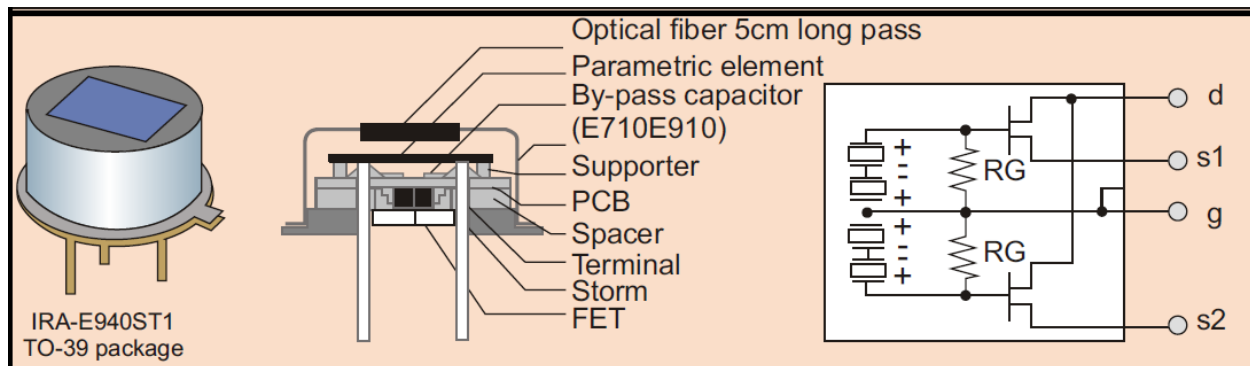


Figure 6.13: Internal structure of PIR sensor [57]

PIR sensor detects infrared energy and so it is sensitive to every live thing that emits infrared energy. Usually PIR detects things that cause sudden change in environments. The sensor is designed in such a way that sensor light turn on when it detects moving object rather than a

standing still object. Because moving object exhibits sudden change in infrared energy than a motionless entity [57].

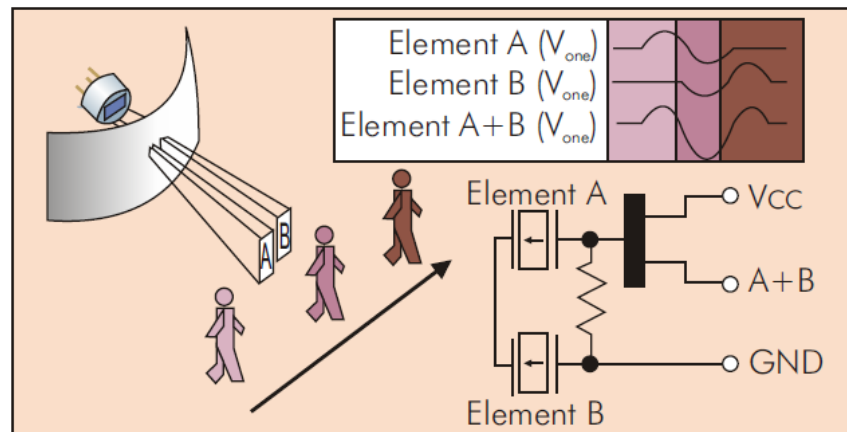


Figure 6.14: PIR sensor operation by sensing different infrared radiation [57]

A dual-element PIR sensor senses variation of normal temperature and invert with respect to each other. Whenever that two inverted elements are in same radiation level they cancel each other out resulting zero voltage meaning movement detected. That corresponds to have a proof of object being present in or between the field of view of the sensor.

6.2.3.2 Effects of atmosphere:

At room temperature (300K) a human emits wavelength of 9600 nm and PIR sensors are designed to work in between 7000 nm to 14000 nm ranges for sensing human. Over most of the temperature range, a PIR sensor works consistently in detecting the temperature difference but at high temperatures, close to body temperature at about 99 degrees F, there is a small range of outside temperatures where a PIR sensor range will reduce.

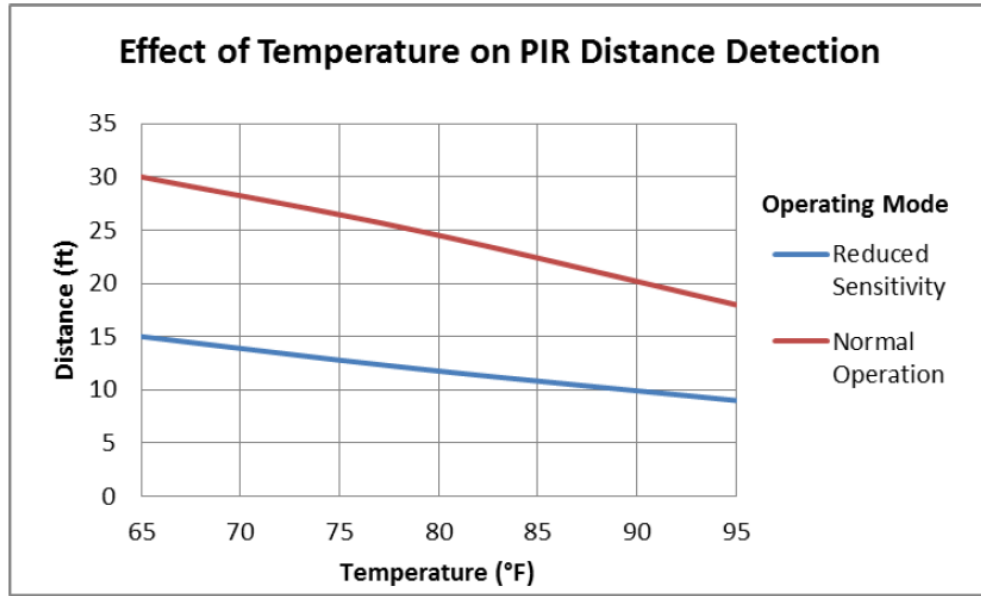


Figure 6.15: Effects of temperature on PIR Sensor [56]

This is because the difference between the background temperature and body temperature is less and the signal generated by the PIR sensor is smaller. When the temperature is higher than around body temperature, the difference starts to be significant again and the range will go back to what is typical.

Object Temperature calculation (based on Stefan-Boltzmann Law) [58]:

$$T_c = \sqrt[4]{T_s^4 + \frac{\phi}{A\sigma\epsilon\epsilon_s}}$$

T_s = sensor's surface temperature

T_c = object's temperature in Kelvin

ϕ = magnitude of net thermal radiation flux

ϵ = emissivity of the object

6.2.3.3 PIR sensor in Flame Detection:

PIR sensor deals with wavelengths of infrared lights. By analyzing light emission of flames a threshold range can be identified for detecting. However, PIR sensors are better compared to smoke or fire detector for its wide range of detection [58].

Chapter -7

Portable Power Supply Design

7.1 Portable Power Supply:

Power sources are an inevitable challenge in the design of portable and small mobile systems, especially robotic systems. Power systems cannot be an afterthought and, if left to final design stages, can result in severe and adverse effects on the system, such as excess weight, size, heat and operational limitations. Power system alternatives include batteries, fuel cells and generators, thermoelectric generators, supercapacitors, flywheels and even non-storage options such as tethers. However, selection and incorporation of power sources should be an early and primary design focus because of its impact on the rest of the system, including size, weight, packaging, mechanism and wiring.

7.1.1 Portable Power Supply for IP Camera

The IP camera used in this project requires 12V/1A constant power supply. So while designing a power supply circuits the requirements are given the most priorities and LM7812 fulfills the criteria.

LM7812:

LM7812 is a three terminal positive regulator with fixed output current and voltage. It has internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



Figure 7.1: Pin Configuration of LM7812

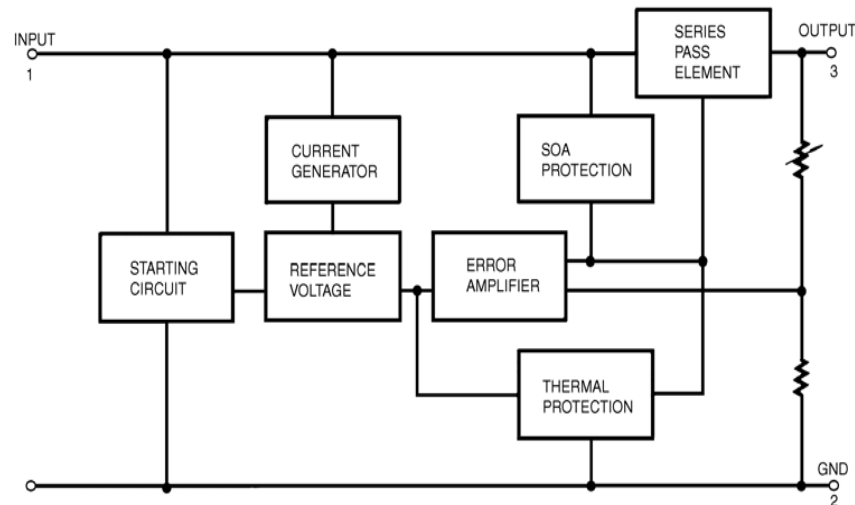


Figure 7.2: Internal circuitry of LM7812

Features:

- Output Current up to 1A
- Output Voltages of 12V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Characteristics:

Voltage regulation means to prevent the voltage from dropping down or rising above a specific value. A 12 volt regulator circuit will provide exactly 12V under load as well as without load. On the other hand, an un-regulated 12V power supply output voltage with drop under while the load will increase and will rise when not under load. For example, an un-regulated 12V power supply can output 14V when not under load and can drop the voltage to 11V when under 1A load and 10V when under 2A load. This rise and drop of voltage in un-regulated power supplies can sometimes burn sensitive electronics equipments [59].

All linear regulators have a minimum dropout voltage, or difference. A 7812 is a linear regulator, and does not step up the input voltage if the input is below the output. The input voltage must be above the output. The maximum dropout voltage V_d is listed as 2.5 volts [60]. Input voltage range is 14V to 35V. 7812 has built in over heat and short circuit protection which makes it a good choice for making power supplies. To achieve 12V 1A current, 7812 should be mounted on a good heat sink plate [61].

Purpose of using LM7812:

To make our IP Camera portable, it is necessary to ensure a constant supplied voltage and current. The required voltage and current for our IP Camera is 12V, 1A. The chips of 78xx group works like a voltage rectangular. All of them ensure a constant voltage supply at loaded and no loaded condition. Among these chips, LM7812 provides constant 12V. It also provides 1A current. Therefore we selected this lm7812 chip to make the camera portable.

Practical Experiments and Graphs:

In practical several voltages are given in the input port of the LM7812 using dc voltage source and output voltage is measured by using multimeter.

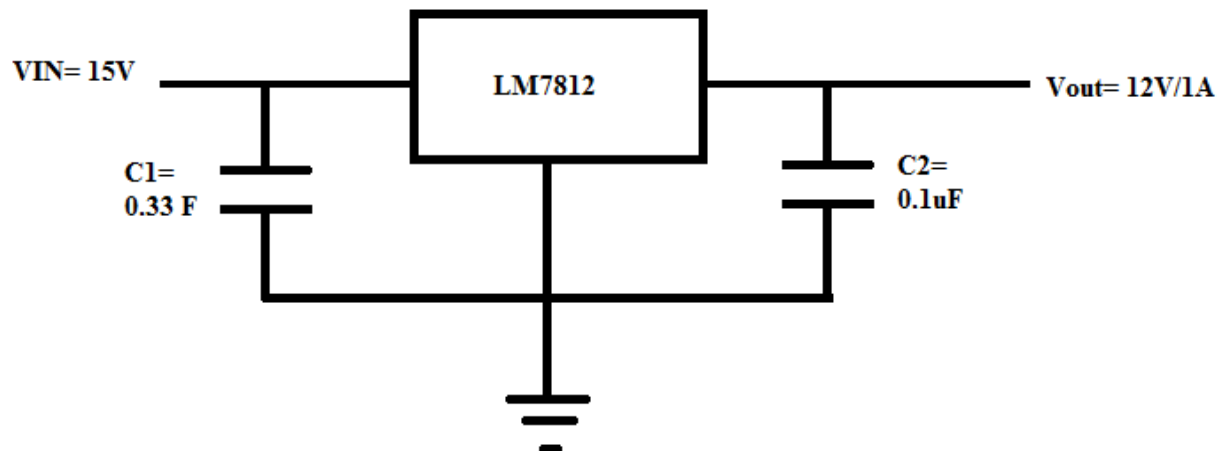


Figure 7.3: Connection of LM7812

Output current is also measured for different input currents. Theses collected datas are plotted using Matlab. The graph shows that the chip gives a constant voltage 12V [Fig.7.5] and constant current 1A [Fig.7.6] in its output which is recurred for our camera.

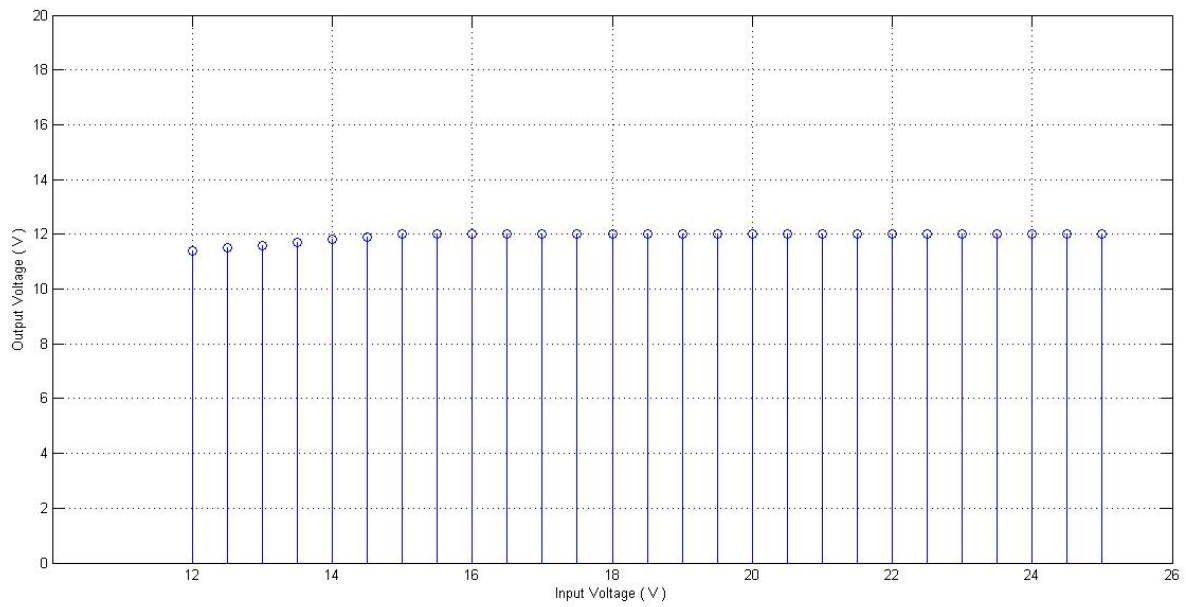


Fig.7.5: Input vs Output Voltage of LM7812

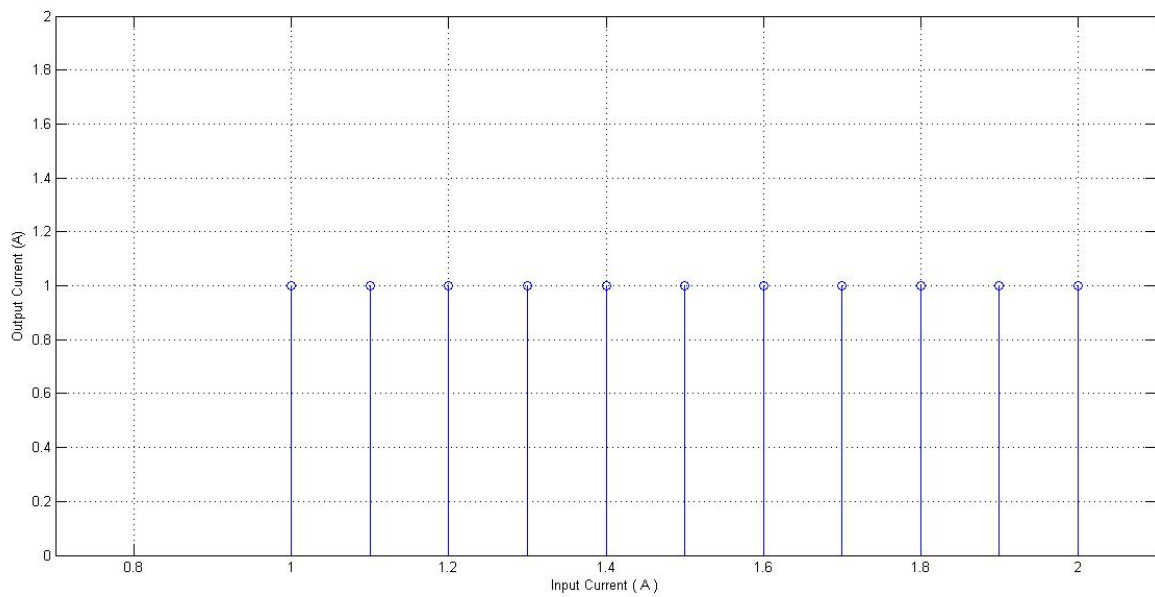


Fig.7.6: Input vs Output current of LM7812

7.1.2 Portable Power Supply for Motor & Arduino

Motors need a lot of energy, especially 12V DC geared motors need 2A current. Therefore 4 motors each needed 2A current resulting 8A only for motors. In that case battery should be selected in such a way that it can provide these current along with supplying current for other electronic devices i.e. sensors, solenoid valves.

In this case, a 12V 18Ah lead acid battery is selected to power up the motors as well as other subsidiary devices.

The microcontroller board Arduino also draws high current above 1.5A because of sensors and wireless modules. So an uninterruptible power source is required to power up the device for 2/3 hours. So, another small 6V 1200mAh battery is adjacent to do that task.

Chapter -8

Software Design

8.1 Software Implementation:

Software implementation of fire extinguisher robot is imperatively important for remote monitoring purposes. The graphical user interface made for the purpose embraces variety of functionality, ease of operability and visual representation.

Motivation behind Software Implementation:

- Less complication in hardware
- Electronics parts alter after using several times need replacement of parts as well as require regular servicing.
- Complication in sensor data viewing in hardware
- GUI represents simplified design of control unit as well as database of sensor values

8.1.1 Introduction to Microsoft Visual Studio 2010:

Microsoft Visual Studio is an IDE from Microsoft that develop console and graphical user interfaces for websites, web applications, and WPF (Windows Presentation Foundation) applications on different platforms. However, it supports Microsoft Windows, Windows Mobile, .NET Framework, Microsoft Silverlight.

Microsoft Visual Studio includes code editor along with debugger and other built-in tools for building GUI applications. In Visual Studio 2010 code editor highlights references and offers quick search feature for symbols in C, C++, C#, VB.NET projects [62].

8.1.2 Control Unit Software:

Control unit software is the central part of the whole structure as it is going to control the movement of robot while under rescue operation. The software is made of Visual studio 2010 software in c# language and is a simple implementation of serial communication. The software simultaneously performs two functions. Firstly the software will act as a client which will send command to its open serial ports that will be used in remote operation of robot from the base station, where on the other hand, software will also receive sensors data sent from the robot to base station by serial communication protocol. Any serial ports can be selected for communications since in the software there is a combo box showing available serial ports.

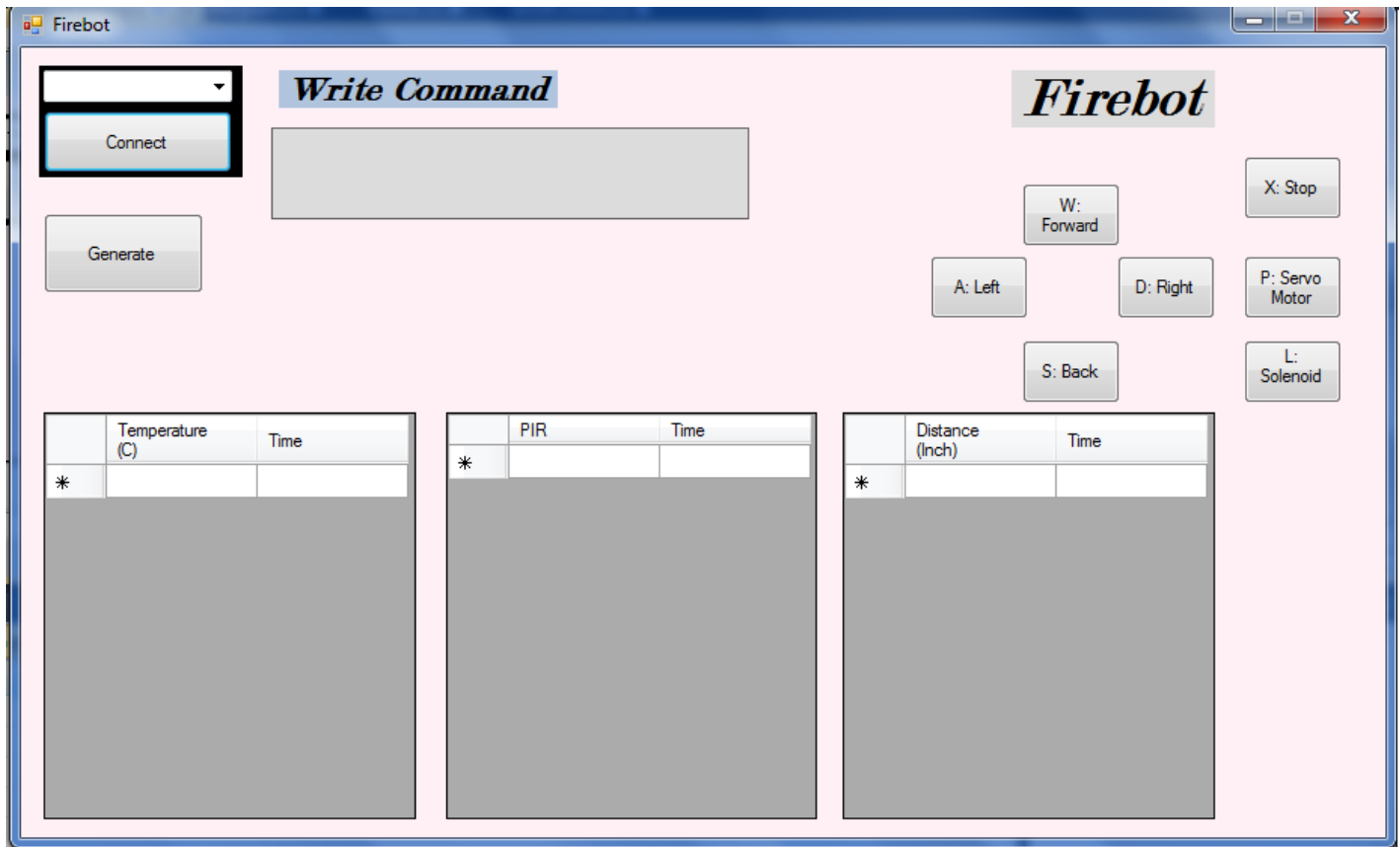


Figure 8.1: Control Unit software

8.1.3 Features of Control Unit Software:

- Simple and easy to use
- No hazard in installation
- Only requirement tools are included
- All-in-One GUI
- Programming language: C++

8.1.4 Functions of Control Unit Software:

- Look for available COM Port
- Connect to Serial COM Port
- Read keyboard input and send through serial port
- Read data from serial port and show in DataGrid

8.2 Block Diagram of Software Operation:

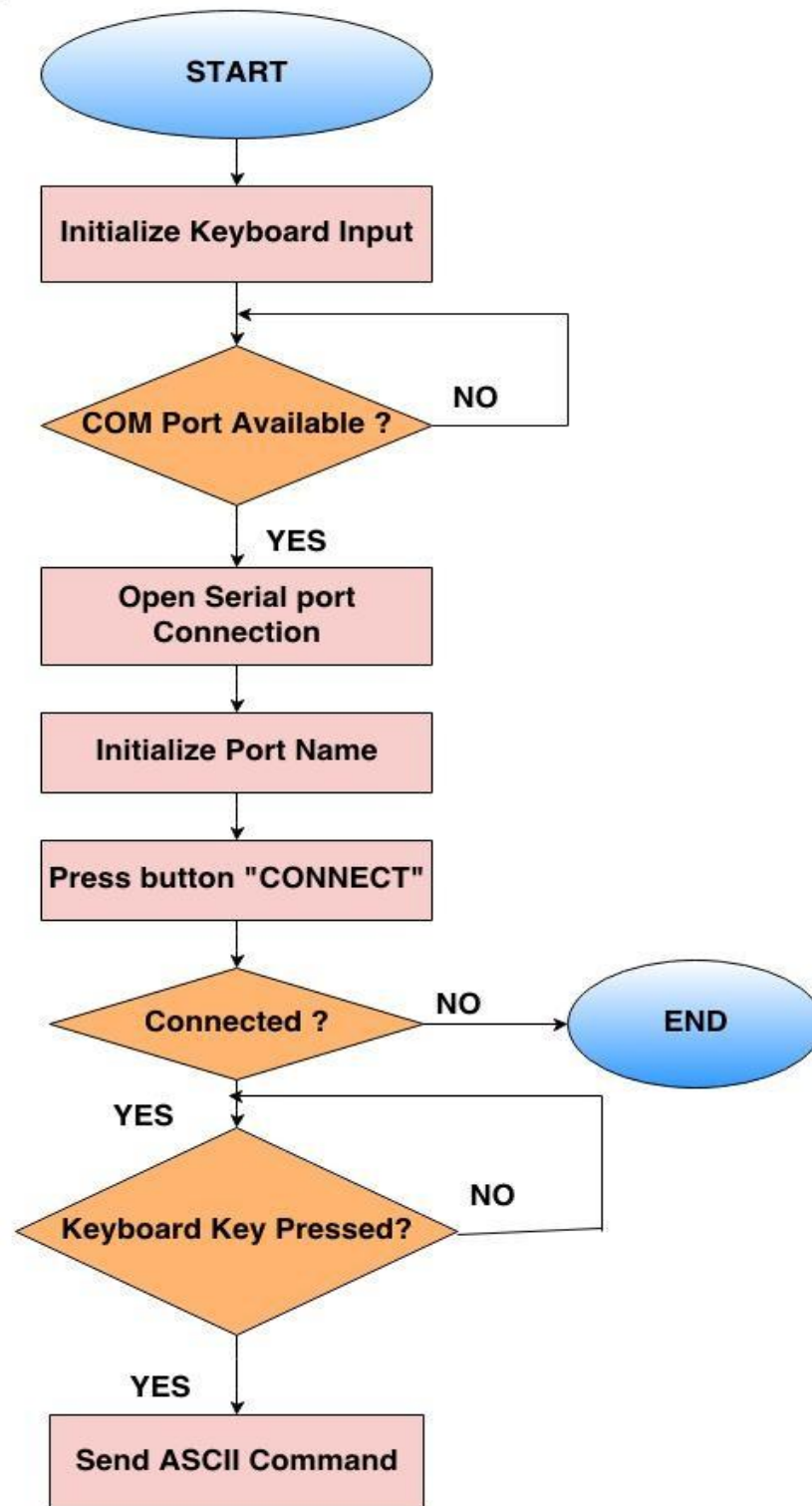


Figure 8.2: Block Diagram for Sending Keyboard Command

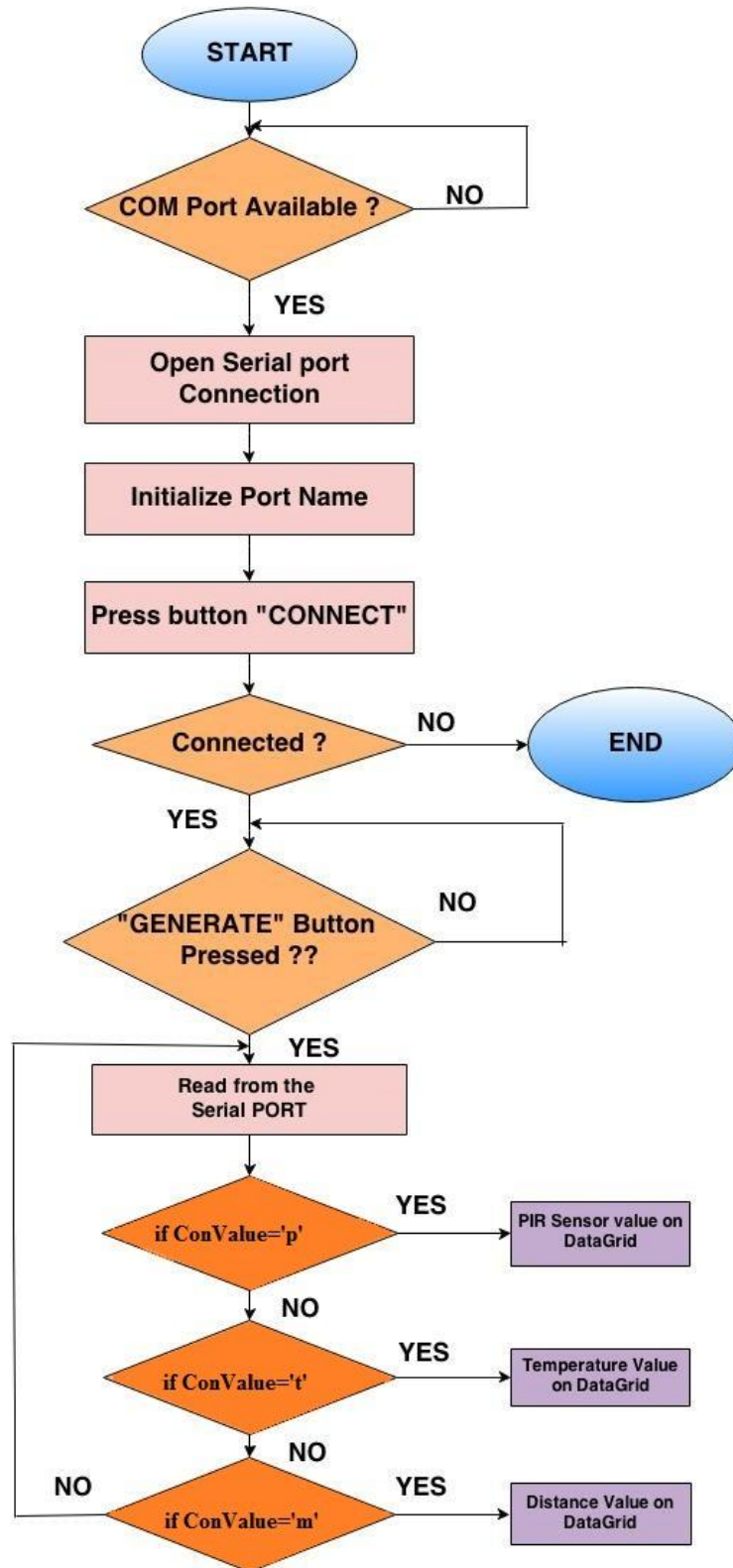


Figure 8.3: Block Diagram of Receiving Sensor data from Serial Port

8.2.1 Block Diagram of Software Interface with Hardware:

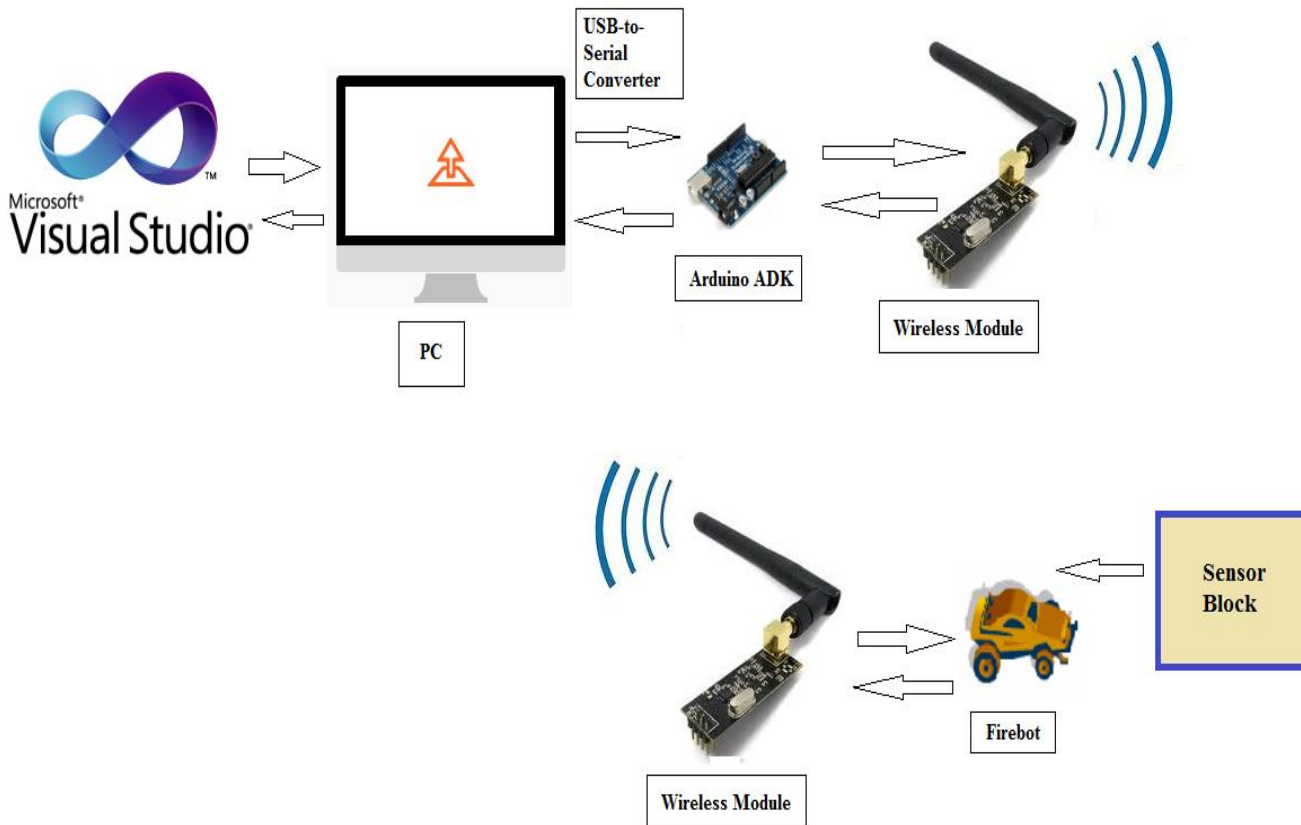


Figure 8.4: Block Diagram of Software and hardware Interface

8.3 Software manual:

8.3.1 General Information:

Installation Procedure:

Double-click on the application and the interface will be prompted.

Purpose:

The purpose of the “Firebot” software is to give an easy access of operator hardware to control the fire extinguisher robot wirelessly and get the sensor data. The software also keep track of the time of received sensor data.

Hardware Requirement:

The hardware requirement of Firebot is:

CPU: Minimum requirement for CPU is a Pentium 1.0 GHz processor. The computer or laptop must have minimum one workable USB port and a Keyboard.

Memory: 512MB of RAM is the minimum requirement

Table 8.1: Keyboard Functions in Firebot Software

Keyboard Input	Function
W	Forward
A	Left
S	Backward
D	Right
X	Stop
P	Servo Motor Control
O	Servo Motor Off
L	Solenoid Valve On
K	Solenoid Valve off

8.3.2 List of User Interfaces:

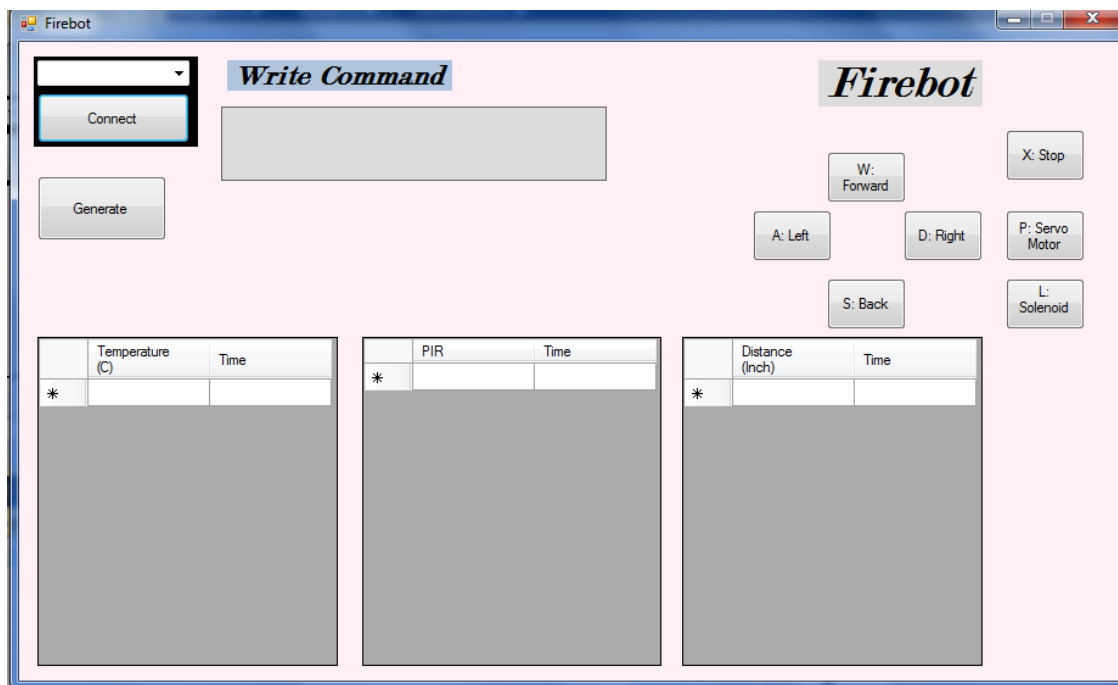


Figure 8.5: User Interface of Firebot

Connect:

After the user identify a com port from the ComboBox, connect button will be pressed.

Write Command:

The space beneath the write command is the place where keyboard input will be taken. User will place the cursor on that corresponding area and press keyboard button. Only selected keyboard key will be accepted.

Generate:

Generate button will be pressed when user is ready to receive sensor data on the data grids.

Data Grids:

The 3 data grids are the place where sensor values will show up.

8.3.3 Getting Started:

Selecting COM Port:

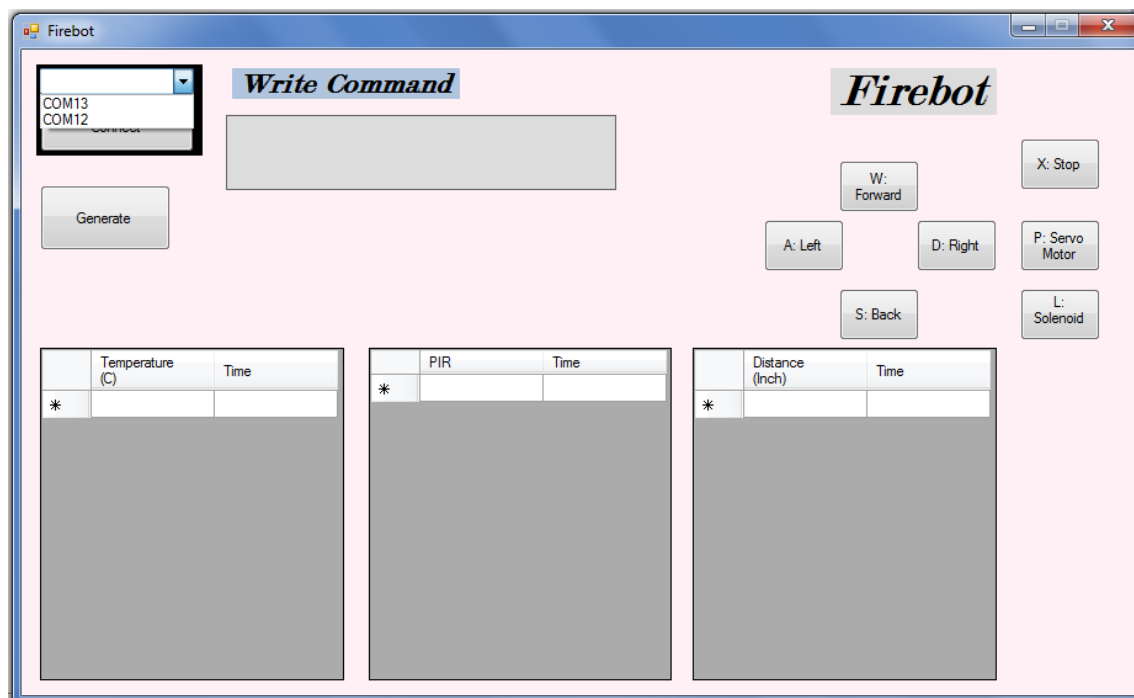


Figure 8.6: Identifying COM Port

Connecting COM Port:

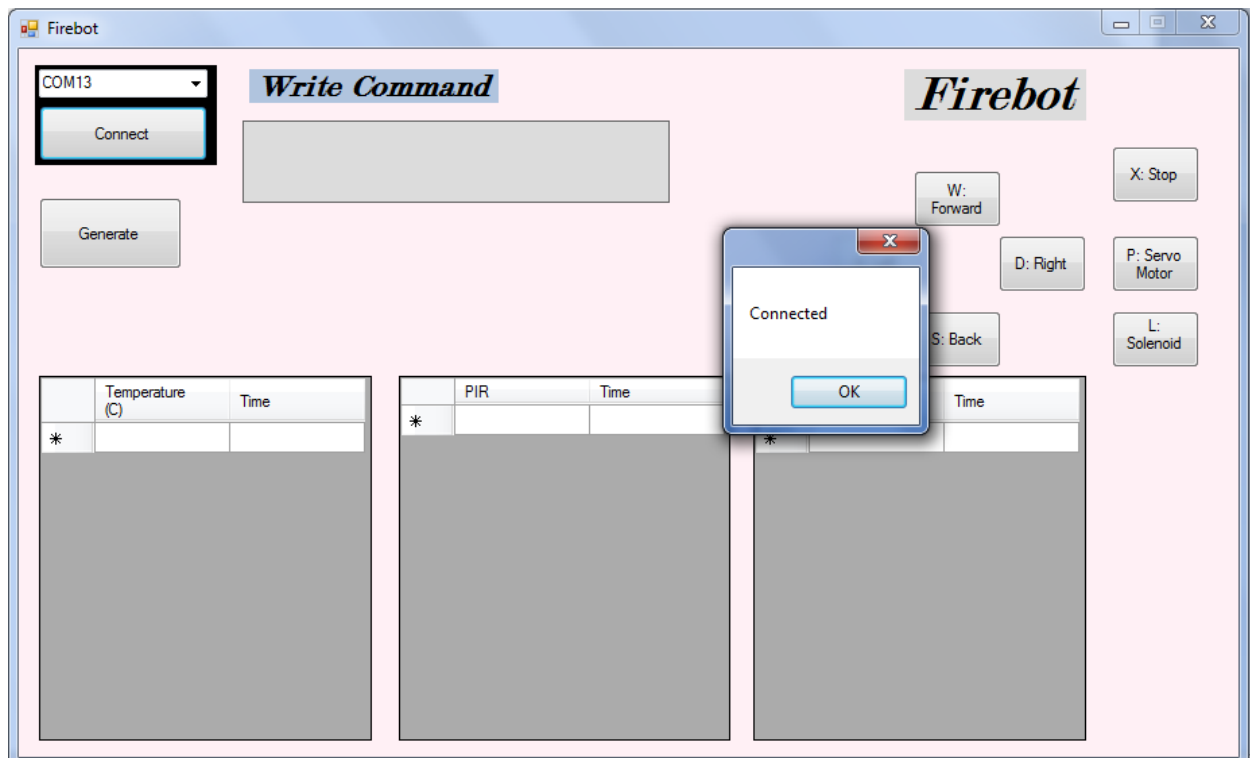


Figure 8.7: Successful Connection of COM Port

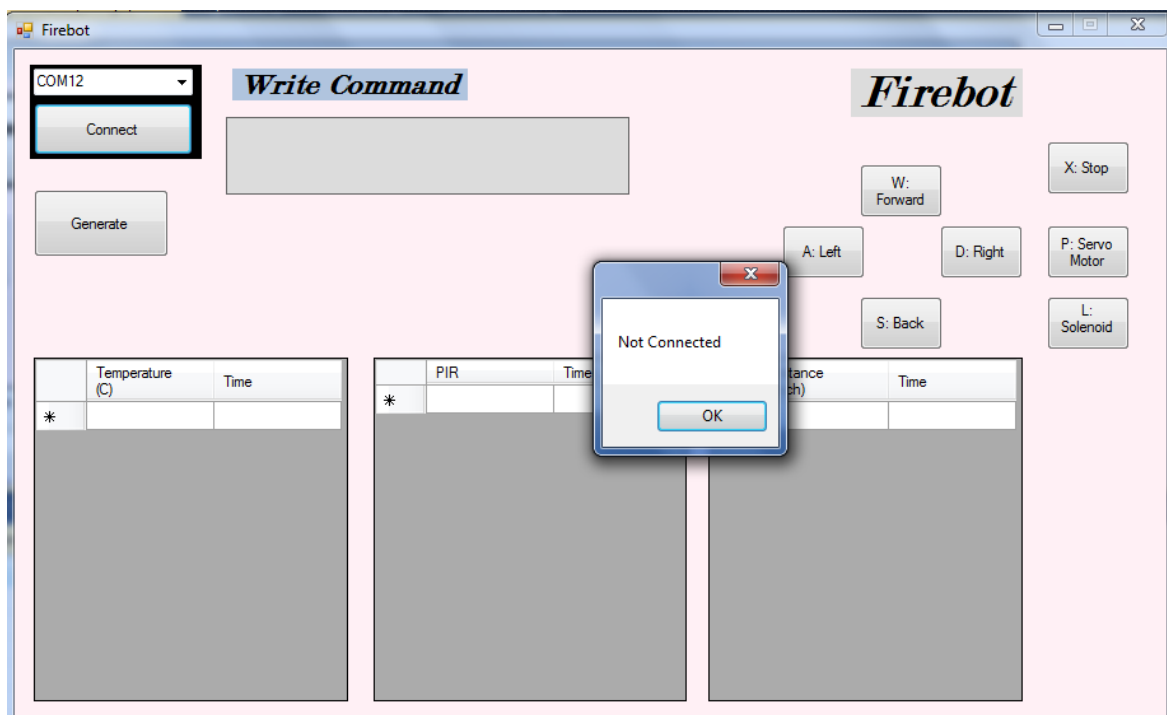


Figure 8.8: Failed in connecting with COM Port

Sending Command:

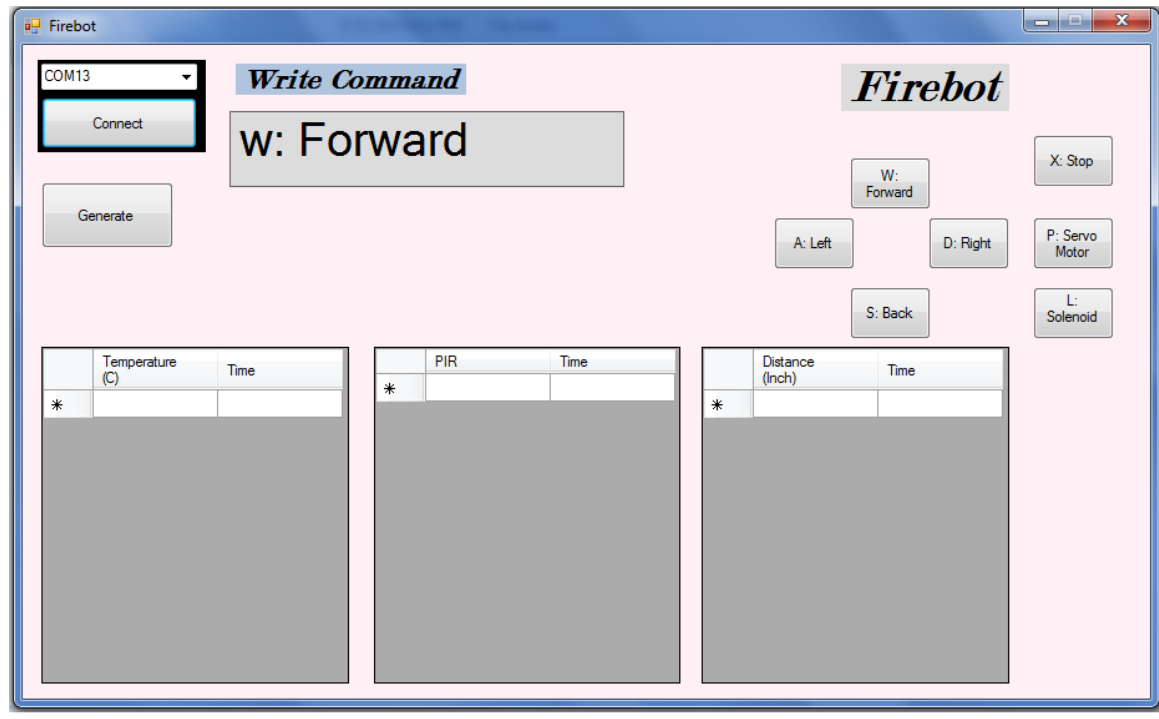


Figure 8.9: Sending “W” command

Generating Sensor Data Table:

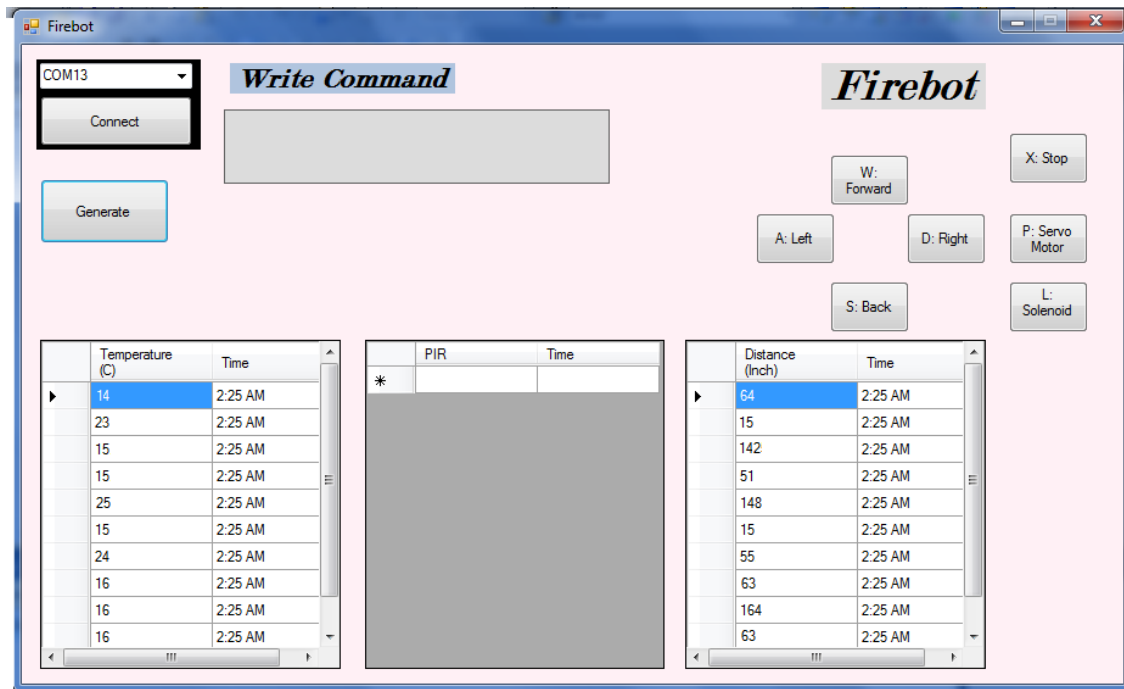


Figure 8.10: Generating Sensor data in Data Grid

Chapter -9

Conclusion

9.1 Summary

The microcontroller based robot with convenient mechanical design and implementation with a view to rescue people from sudden rapid oxidation resulting exothermic chemical process of combustion is the Fire Fighting Robot. In short, the main objective of the developed unique Fire Fighting Robot is not only to extinguish rapid oxidation which produces exothermic heat but also to monitor any hazardous place where human fire fighters would face harder to move into it. On that important aspect it is necessary to have an efficient design. An efficient design would never be efficient until the design is virtually simulated in computer software. In this case, our designed firefighting robot's computer aided design was prepared graphically through computer and then simulated to identify different problems of it. Here, the problems which were found never had exact solutions rather it demanded modifications and consequently it were all done. Starting from scrap paper and pencil to the final implementation in real life, the whole design was modified for several times. Obviously while designing this fire extinguishing robot, we had an intention to increase mobility, responsiveness and stability. To increase this mobility, responsiveness and stabilization the robot has been made into four wheeled vehicle shaped. It was experimentally observed that, if the robot's main movement could be controlled by primary 4 motors, then responsiveness would never be compromised rather it will increase mobility as well. As the robot is intended to response in a very hazardous environment, so the stability was one of the important concerns. The surrounding temperature has always been a negative factor working on robot's body and chamber, so to save the internal circuitry from excessive heat it is required to cool down the chamber's temperature level. A three blade fan is continuously rotated inside chamber to control temperature. While designing, this cooling system was also considered, so the design was proven to be stable enough. Only temperature cooling could never make the design efficient, until it was not chosen which materials to be used to build external and internal body shape. The bonnet and hood of FireBOT were designed with steel and aluminum respectively because the melting point was higher than any other cheap materials. As the melting point was high, so it was used to build the FireBOT which is proposed to be sent into any hazardous situation where fire swept away in everywhere.

Manual mode of operations is user dependent by providing command to the FireBOT's microcontroller wirelessly. The chip which is used for wireless communication is named as nrF24l01+LNA+PA chip, which has the lowest noise level including higher RF power amplification ability. In manual controlled mode, the user can control FireBOT on the basis of visual data reception. To send video data stream from FireBOT to remote server, an IEEE 802.11 wireless LAN protocol based Wi-Fi technology dominated IP Camera is used which has the best quality of transferring video in a moderate distance. According to the video data reception the user could move FireBOT in any direction where is required to move. Just moving from one direction to another direction is the sole purpose of FireBOT. The robot can also be used to do surveillance purpose because sometimes the situation gets such worsen that human fire fighter cannot enter into that place because of dense fire swept heavily in everywhere.

In manual control mode, four functions are available. They are:

1. Controlling FireBOT's main motors
2. Movement of nozzle to target for extinguishing fire
3. Command for going into auto-pilot mode
4. Graphical and numerical representation of different sensors value

The above functions are executed by providing command from a GUI, which is programmed and designed only for the purpose of controlling FireBOT. The graphical user interface (GUI) is used to view the video streaming data along with the value of sensors. A user can take decision where to move and open solenoid valve to remove fire extinguishing material basing on visual data reception. FireBOT works on the basis of artificial intelligence. The microcontroller using the help of five kinds of sensors, FireBOT takes decisions where to move, in which things to spray extinguishing materials and how to get safely return from that dangerous place.

9.2 Final Remark:

In conclusion, our approach of designing a unique vision of concepts in fire extinguishing robot promotes reliable and efficient outcome from various electrical equipments. The extensive use of embedded system ensures the integration with flexibility in control and operation. There were still a number of problems step up in the stage of integration however by debugging the circuitry a way of solution is found out. This thesis report presents the major features and functions of each block of circuitry to ensure proper reliability. The report also states a detailed design and mechanism of each part of the robot that continuously monitors, communicates and extinguish the fire.

9.3 Future scope:

This project is motivated to detect fires and take appropriate steps without any human intervention. In the present condition it can be operate through wireless command with the help of visual data sent by the wi-fi enable IP camera. Also sensor data helps the operator to presume the condition of that affected area by sending the values. A real life dry ice based two extinguisher is also set to extinguish. However, there are opportunities of modifying the design to get acceptable with newer technology. Some future plans are:

1. Autonomous Fire extinguisher robot: Along with the manual mode, we give importance of autonomous mode for having a better output. Whenever the fire affected area covered with the dense smoke, camera could send clear video. On that situation, activation of autonomous mode will allow robot to be dependent of its high powered sensors for path navigation and proper extinguish of robot.

2. Android-controlled fire extinguisher robot: Since in this project Arduino ADK is used in robot part, it can be modified in the level of android control. Software has to bring to operate it.

References:

- [1] Correspondent, Special. "TERRIFIC TRAGEDY IN CHILI." *The New York Times* [New York] 18 Jan. 1864: pg 1 , <http://www.nytimes.com/1864/01/18/news/terrific-tragedy-chili-two-thousand-five-hundred-persons-roasted-death-church.html>
- [2] PTL. "Major fire at Hyderabad hospital." *DNAindia* [India] Tuesday, Feb 2, 2010 : <http://www.dnaindia.com/india/1342441/report-major-fire-at-hyderabad-hospital-one-patient-dead>
- [3] Duell ,Mark. Reilly, Jill. "FIFTY DEAD...and it's NOT over", *mailOnline* [USA], 30 October 2012 : <http://www.dailymail.co.uk/news/article-2225108/Hurricane-Sandy-2012-pictures-FIFTY-DEAD--NOT-over.html>
- [4] The free encyclopedia.(2011, April 3). FL: Wikimedia Foundation, Inc. retrieved 18 December2013, from: http://en.wikibooks.org/wiki/Robotics/Design_Basics/Building_Materials
- [5] Flether. "The Effects of High-Pressure, Hydrogen on steel", March 24, 1964.
- [6] The free encyclopedia. (2009,May 30). FL: Wikimedia Foundation, Inc. retrieved 18 December, From:<http://www.pilkington.com/products/bp/bybenefit/fireprotection/pyrostop/default.htm>
- [7] Ron, C. "Physical Properties, Characteristics and Alloys".p-13
- [8] Arduino, Wikipedia, [Online] 2013, <http://en.wikipedia.org/wiki/Arduino> (Accessed : 20 November 2013)
- [9] Arduino ADK, Arduino, [Online], <http://arduino.cc/en/Main/ArduinoBoardADK> (Accessed : 20 November 2013)
- [10] M. McRoberts, *Beginning Arduino: Technology in action*, Apress, 2010. [Online] Available: Google Books
- [11] Arduino Mega 2560, Arduino, [Online], <http://arduino.cc/en/Main/arduinoBoardMega2560> (Accessed: 20 November 2013)
- [12] Isocom Components, "Optically Coupled Isolator Phototransistor Output," 4N35 datasheet
- [13] A. Malvino, *Electronic Principles*, 7th ed. Reading, Tata McGraw-Hill Education, 2007. [E-book] Available: Google books
- [14] P. Bergsman, *Controlling the World with Your PC*, Newnes, 1994.[E-book] Available: Google Books
- [15] International IOR Rectifier, "IRFP250N HEXFET Power MOSFET," IRFP250N datasheet
- [16] International Rectifier, Appl. Note, Power MOSFET Basics, pp 6-7
- [17] Alpha & Omega Semiconductors, Appl. Note-Power MOSFET Basics, pp 2-3
- [18] The New York Times, December 8, 1863, from: <http://www.newyorkirishcenter.org/>
- [19] The free encyclopedia (2004, July 22). FL: Wikimedia Foundation, Inc. retrieved 11 November 2012,from: <http://www.fireservice.gov.bd/>
- [20] Fire Death Rate Trends: An International Perspective (2011), retrieved from July 2011, from: <http://www.usfa.fema.gov/downloads/pdf/statistics/v12i8.pdf>
- [21] Yan Zhou, Dawn Wilkins, Robert P. Cook. Neural Network Control for A Fire-Fighting Robot. University of Mississippi, Dept. of Computer and Information Science, Weir 302, University, MS 38677;
- [22] Aman Sharma. A FULLY AUTOMATED FIRE FIGHTING ROBOT. Studying engineering in Dronacharya College of Engineering Haryana, India
- [23] A. Hand. *Electric Motor Maintenance and Troubleshooting*, 2nd ed. Reading, McGraw Hill Professional, 2011. [E-book] Available: Amazon

- [24] V. Subramanian, M. Kaye, P. Chopra. ESE 206. Class lecture, Topic: "H-Bridge Motor Control." Department of Electrical and System Engineering, University of Pennsylvania, Philadelphia, PA. Mar. 2012
- [25] B. O'Donnell. PHYS 483. Class lecture, Topic: "Relay Basics." Department of Physics and Astronomy, University of Nevada Las Vegas, Las Vegas, LV.
- [26] International IOR Rectifier, "IRFP250N HEXFET Power MOSFET," IRFP250N datasheet
- [27] A.S. Sedra, K.C. Smith. Microelectronics Circuits, 5th ed. Reading, MA: Oxford University Press Inc., 2009.
- [28] G. Recktenwald. EAS 199. Class lecture, Topic: "Basic PWM Properties." Maseeh College of Engineering & Computer Science, Portland State University, Portland, OR. Fall 2011.
- [29] Technopedia. "Wireless Communications". Internet: <http://www.techopedia.com/definition/10062/wireless-communications>. [November 15, 2013].
- [30] M. Rouse. (April 2006). "Tech Target - What is wireless?" Internet: <http://searchmobilecomputing.techtarget.com/definition/wireless>, [December 2, 2013]
- [31] Pahlavan, Kaveh, Levesque, Allen H (1995). *Wireless Information Networks*. John Wiley & Sons. ISBN 0471106070.
- [32] Aman Mangal. (June 6, 2012). " Serial Communication between Arduino and MATLAB". IITBombay.[On-line]. Available: http://home.iitb.ac.in/~rahul/ITSP/serial_comm_matlab.pdf
- [33] SparkFun Electronics. (2012). Introduction to Arduino. Available: http://dlnmh9ip6v2uc.cloudfront.net/learn/materials/1/Arduino_final_handout.pdf
- [34] Elec Freaks. 2.4G Wireless nRF24L01p with PA and LNA. Available: http://www.elec Freaks.com/wiki/index.php?title=2.4G_Wireless_nRF24L01p_with_PA_and_LNA
- [35] Valentin Dzhankhotov . (2009). " HYBRID LC FILTER FOR POWER ELECTRONIC DRIVES: THEORY AND IMPLEMENTATION ". [On-Line]. Available: <http://www.doria.fi/bitstream/handle/10024/47410/isbn%209789522148278.pdf>
- [36] MikroElektronika." Digital Filter Design" . Available: <http://www.mikroe.com/chapters/view/71/>
- [37] Wikipedia. Phase-shift keying. Available: http://en.wikipedia.org/wiki/Phase-shift_keying
- [38] Clifford E. Cummings, Peter Alfke. (2002). " Simulation and Synthesis Techniques for Asynchronous FIFO Design with Asynchronous Pointer Comparisons". SunBurst Design. Available: http://www.sunburst-design.com/papers/CummingsSNUG2002SJ_FIFO2.pdf
- [39] Kuo-Jung Sun, Zuo-Min Tsai, Kun-You Lin, "A Noise Optimization Formulation for CMOS Low-Noise Amplifiers With On-Chip Low-Q Inductors " in IEEE Transactions on Microwave Theory and Techniques, Vol 54, NO 4. Feb 2006.
- [40] FREQUENCY SYNTHESIZER U.S. Patent 3,555,446, Braymer, N. B., (1971, January.

- [41] Ulrich L. Rohde "Digital PLL Frequency Synthesizers - Theory and Design ", Prentice-Hall, Inc., Englewood Cliffs, NJ, January 1983
- [42] nRF24L01+ Single Chip 2.4GHz Transceiver, Product Specification v1.0. (September 2008). NORDIAC SEMICONDUCTOR, DATASHEET. Available: <http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01>
- [43] Tronixstuff. Tutorial: Arduino and the SPI bus. Available at: <http://tronixstuff.com/2011/05/13/tutorial-arduino-and-the-spi-bus/>
- [44] The Philadelphiafire code, chapter 900- Fire Protection System.
- [45] *The free encyclopedia*. (2003, July 22). FL: Wikimedia Foundation, Inc. Retrieved 13December 2013 , from: <http://www.ch.cam.ac.uk/safety/fire-safety-types-extinguisher>
- [46] James, G.B.(1920). "Design of a foam fire extinguisher system".
- [47] Tehrani, M. "Pressure Control of a Pneumatic Actuator ", May 2008
- [48] *The free encyclopedia*. (2004, July 22). FL: Wikimedia Foundation, Inc. Retrieved August 10, 2004, from: http://en.wikipedia.org/wiki/Solenoid_valve
- [49] 2006, Nelson Publishing, 2008, Gale, Cengage Learning.
URL:http://findarticles.com/p/articles/mi_m0CMN/is_1_43/ai_n26771543
- [50] *The free encyclopedia*. (2004, July 22). FL: Wikimedia Foundation, Inc. Retrieved 11December 2013 , from: http://en.wikipedia.org/wiki/IP_camera
- [51] Matthews, D., "Choosing and Using a Temperature Sensor," *Sensors Magazine*, Vol. 17, No. 1, January 2000.link: <http://archives.sensorsmag.com/articles/0100/54/index.htm>
- [52] Scolio, J., "Temperature Sensor IC's Simplify Designs," *Sensors Magazine*, Vol. 17, No. 1, January 2000. Link: <http://archives.sensorsmag.com/articles/0100/48/index.htm>
- [53] Texas Instruments, "LM35 Precision Centigrade Temperature Sensors," LM35 datasheet, Aug. 1999 [Revised Oct. 2013].
- [54] Audiowell Electronics, "Ultrasonic Electronic Eye Telemeter Module," TS601 datasheet.Link: <http://www.slideshare.net/onlinetps/ts601p01-ultrasonic-distance-modulepdf>
- [55] J. David and N. Cheeke, "Fundamentals of ultrasonic waves," CRC Press, Florida, USA, 2002, ISBN 0-8493-0130-0.
- [56] Parallax Inc., "PIR Sensor," PIR sensor module datasheet, Feb. 2007.
- [57] K. Kohli,"PIR sensor in Motion Detection & Identification System," *electronicsmaker.com*, [Online]. Available: <http://www.electronicmaker.com/em/admin/pdf/free/SENSOR.pdf>. [Accessed Nov. 25, 2013]
- [58] B. Saracoglu. ECE 480. Class lecture, Topic: "A Guide to IR/PIR Sensor Set-Up and Testing." School of Electrical and Computer Engineering, Michigan State University.

[59] Research Cell, Voltage Regulator Circuit; from:
<http://www.researchcell.com/electronics/voltage-regulator-circuit/>

[60] Electrical Engineering Stack Exchange is a question and answer site for electronics and electrical engineering professionals, students, and enthusiasts. From:
<http://electronics.stackexchange.com/questions/7463/7812-minimum-input-voltage>

[61] Research Cell, 7812 Pin and Circuit Diagram from:
<http://www.researchcell.com/electronics/7812-pin-and-circuit-diagram/>

[62] S. Somasegar, "Code Focused Development in VS Studio," *blogs.msdn.com*. [Online]
Available:<http://blogs.msdn.com/b/somasegar/archive/2008/12/19/code-focused-development-in-vs-2010.aspx> [Accessed: 27 November 2013]

Appendix

Codes for the operator Hardware:

```
#include <SPI.h>

#include <Mirf.h>

#include <nRF24L01.h>

#include <MirfHardwareSpiDriver.h>

int rate,k;

void setup(){

  Serial.begin(9600);

  Mirf.spi = &MirfHardwareSpi;

  Mirf.init();

  Mirf.setRADDR((byte *)"clie1");

  //Mirf.payload = sizeof(rate);

  Mirf.payload=1;

  Mirf.channel=90;

  Mirf.config();

}

void loop(){

  if (Serial.available()){

    int msg=Serial.read();

    Serial.println(msg);

    Mirf.setTADDR((byte *)"serv1");

    Mirf.send((byte *) &msg);

    while(Mirf.isSending()&&!Mirf.dataReady()){
```

```

    Serial.println("sending");
}

}

//----- sensor value receive-----

byte c;

// is there any data pending?
if( Mirf.dataReady() )
{
    // well, get it
    Mirf.getData((byte*)&c);
    // Serial.println("got packet");
    // ... and write it out to the PC
    Serial.print((char)c);
    if((char)c=='m'){
        Serial.println();
    }
    if((char)c=='t'){
        Serial.println();
    }
    delay(300);
}
}

```

Codes for the Fire Extinguisher Robot part:

```

// robot part

#include <SPI.h>

#include <Mirf.h>

```

```

#include <nRF24L01.h>

#include <MirfHardwareSpiDriver.h>

const int pingPin = 12; // Sonar pin

int pirPin = 31; // pir pin

int LedPin = 53; // for PIR sensor

int msg;

int motor_pins[]={2,3,4,5,6,9,10,11};

int pincount=8;

void setup(){

  Serial.begin(9600);

  for (int thispin=0;thispin<pincount;thispin++){

    pinMode(motor_pins[thispin],OUTPUT);

  }

  pinMode(24, OUTPUT);

  pinMode(26, OUTPUT);

  pinMode(pirPin, INPUT); // pir pin will take input

  Mirf.spi = &MirfHardwareSpi;

  Mirf.init();

  Mirf.setRADDR((byte *)"serv1");

  Mirf.setTADDR((byte *)"clie1");

  //Mirf.payload = sizeof(msg);

  Mirf.payload=1;

  Mirf.channel=90;

  Mirf.config();

  Serial.println("ready");

}

```

```

void loop(){

while(!Mirf.dataReady() && Mirf.isSending()){

    Serial.println("no packet");

    delay(500);

}

if (Mirf.dataReady()){
Mirf.getData((byte *) &msg);

Serial.print("keyboard input=");

Serial.println(msg);

//Serial.println("%");

delay(500);

//----- Motor -----

if (msg==119){// Forward movement-- 'w' is pressed

digitalWrite(motor_pins[0],LOW);//pin2

digitalWrite(motor_pins[1],HIGH);//pin3

digitalWrite(motor_pins[2],HIGH);//pin4

digitalWrite(motor_pins[3],HIGH);//pin5

digitalWrite(motor_pins[4],LOW);//pin6

digitalWrite(motor_pins[5],HIGH);//pin9

digitalWrite(motor_pins[6],HIGH);//pin10

digitalWrite(motor_pins[7],HIGH);//pin11

}

if (msg==115){//Backward movement-- 's' is pressed

digitalWrite(motor_pins[0],HIGH);//pin2

digitalWrite(motor_pins[1],HIGH);//pin3

digitalWrite(motor_pins[2],LOW);//pin4

```

```

digitalWrite(motor_pins[3],HIGH);//pin5
digitalWrite(motor_pins[4],HIGH);//pin6
digitalWrite(motor_pins[5],HIGH);//pin9
digitalWrite(motor_pins[6],LOW);//pin10
digitalWrite(motor_pins[7],HIGH);//pin11
}

if (msg==97){ //Left Turn movement-- 'a' is pressed
for(int pins=0;pins<pincount;pins++){
digitalWrite(motor_pins[pins],HIGH);
}
}

if (msg==100){ //Right Turn movement-- 'd' is pressed
digitalWrite(motor_pins[0],LOW);//pin2
digitalWrite(motor_pins[1],HIGH);//pin3
digitalWrite(motor_pins[2],LOW);//pin4
digitalWrite(motor_pins[3],HIGH);//pin5
digitalWrite(motor_pins[4],LOW);//pin6
digitalWrite(motor_pins[5],HIGH);//pin9
digitalWrite(motor_pins[6],LOW);//pin10
digitalWrite(motor_pins[7],HIGH);//pin11
}

if (msg==120){ // Stop all the motors.-- 'x' is pressed
for(int pins=0;pins<pincount;pins++){
digitalWrite(motor_pins[pins],LOW);
}
}

if (msg==112){ // p=servo motor on

```

```

digitalWrite(26, HIGH);

}

if (msg==111){ // o=servo motor off
digitalWrite(26, LOW);

}

if (msg==108 ){ // l= solenoid valve ON
digitalWrite(24, HIGH);

}

if (msg==107 ){ // k= solenoid valve ON
digitalWrite(24, LOW);

}

}

//----- Motor -----
//----- transmit-----

long duration;

float inch, cm;

pinMode(pingPin, OUTPUT);

digitalWrite(pingPin, LOW);

delayMicroseconds(2);

digitalWrite(pingPin, HIGH);

delayMicroseconds(500);

digitalWrite(pingPin, LOW);

pinMode(pingPin, INPUT);

duration = pulseIn(pingPin, HIGH);

// convert the time into a distance

```



```

    inch = microsecondsToInches(duration);

    Serial.print(inch);

    Serial.print(" inch");

    Serial.println();

    transmit("m");

    // transmit the first value

    transmit(inch);

// transmit a CR/LF for the PC

    // software to sync to

    transmitlf();

    // ... just take your time

    delay(1000);

//----- temperature-----

    int tempValue = analogRead(A5);

    int temperature =tempValue*0.488;

    Serial.print("temp= ");

    Serial.println(temperature);

    transmit("t");

    transmit(temperature);

    transmitlf();

//----- PIR sensor-----

    pinMode(LedPin,OUTPUT);

    int pirVal = digitalRead(pirPin);

    if(pirVal == LOW){

        digitalWrite(LedPin,HIGH);

        Serial.println("Human Detected");

```

```

    transmit("p");

    transmitlf();

    delay(1000);

}

else{

    digitalWrite(LedPin,LOW);

    Serial.println("No civilian");

}

    //----- PIR sensor-----

}

long microsecondsToInches(long microseconds)

{

    return microseconds / 74 / 2;

}

void transmit( float v)

{

    byte c;

    char buf[5];

    Serial.print("v=");

    Serial.println(v);

    // dtostrf(FLOAT,WIDTH, PRECISION,BUFFER)

    dtostrf(v,5,2,buf);

    ///long to array

    //ltoa(v,buf,2);

    Serial.print("buf=");

    Serial.println(buf);

```

```

for( int i=0 ; i<4 ; i++ )
{
    c = buf[i];
    Serial.print("c=");
    Serial.println(c);
    Mirf.send(&c);
    while( Mirf.isSending() ) ;
}
// Serial.println(buf);
}
// sends a string via the nRF24L01
void transmit(const char *string)
{
    byte c;
    for( int i=0 ; string[i]!=0x00 ; i++ )
    {
        c = string[i];
        Mirf.send(&c);
        Serial.println(c);
        while( Mirf.isSending() ) ;
    }
}
// send a CR/LF sequence via the nRF24L01
void transmitlf(void)
{
    byte c;

```

```

//\r (Carriage Return) - moves the cursor to the beginning of the line without advancing to the next line
c = 'r';
Serial.print((char)c);
Mirf.send(&c);
while( Mirf.isSending() ) {
}

//\n Line Feed
c = 'n';
Mirf.send(&c);
Serial.print((char)c);
while( Mirf.isSending() ) {
}
}

```

Software Code:

Program.cs:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;
namespace Humidity
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}

```

SerialComm.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.IO.Ports;
namespace Humidity
{
    public class SerialComm
    {
        public SerialPort serialPort;
        public String[] portNames;
        string currentPortName;
        public SerialComm()
        {
            serialPort = new SerialPort();
            portNames = SerialPort.GetPortNames();
        }

        public void reconect()
        {
            connect(currentPortName);
        }
        public bool connect(String portName)
        {
            try
            {
                if (!serialPort.IsOpen)
                {
                    serialPort.PortName = portName;
                    currentPortName = portName;
                    serialPort.Open();
                    return true;
                }
                else
                {
                    return false;
                }
            }
            catch (Exception)
            {
                disconnect();
                return false;
            }
        }

        public string receieve()
        {
            string output = "";
            if (serialPort.IsOpen)
            {
                output = serialPort.ReadLine();
            }
            return output;
        }
    }
}
```

```

        public string send(String command)
        {
            try
            {
                if (serialPort.IsOpen)
                {
                    serialPort.Write(command);
                    return "Sending: " + command;
                }
            }
            catch (Exception)
            {
                disconnect();
                recoonect();
            }

            return "Not Sending";
        }
        public void disconnect()
        {
            if (serialPort.IsOpen)
            {
                serialPort.Close();
                serialPort.Dispose();
            }
        }
    }
}

```

Form1.cs:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows;
using System.Data;
using System.Windows.Forms;

namespace Humidity
{
    public partial class Form1 : Form
    {
        #region Variables
        SerialComm serialComm;
        #endregion
        string RxString;
        public Form1()
        {
            InitializeComponent();
            this.KeyPreview = true;

```

```

        this.KeyUp += new KeyEventHandler(Form1_KeyUp);
        serialComm = new SerialComm();
        initPorts();
    }

    private void Form1_Load(object sender, EventArgs e)
    {

    }

    void initPorts()
    {
        for (int i = 0; i < serialComm.portNames.Length; i++)
        {
            portsCombo.Items.Add(serialComm.portNames[i]);
        }
    }

    private void portsCombo_SelectedIndexChanged(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        if (serialComm.connect((String)portsCombo.SelectedItem))
        {
            MessageBox.Show("Connected");
        }
        else
        {
            MessageBox.Show("Not Connected");
        }
    }

    void Form1_KeyUp(object sender, KeyEventArgs e)
    {
        if (e.KeyCode == Keys.W)
        {
            label12.Text = "w: Forward";
            serialComm.send("w");
        }
        else if (e.KeyCode == Keys.S)
        {
            label12.Text = "s: Reverse";
            serialComm.send("s");
        }
        else if (e.KeyCode == Keys.A)
        {
            label12.Text = "a: Left";
            serialComm.send("a");
        }
        else if (e.KeyCode == Keys.D)
        {
            label12.Text = "d: Right";
            serialComm.send("d");
        }
        else if (e.KeyCode == Keys.X)
    }

```

```

        {
            label2.Text = "X: Stop";
            serialComm.send("x");
        }
        else if (e.KeyCode == Keys.L)
        {
            label2.Text = "L: Solenoid Valve";
            serialComm.send("l");
        }
        else if (e.KeyCode == Keys.P)
        {
            label2.Text = "P: Servo Motor";
            serialComm.send("p");
        }
    }

    private void button2_Click(object sender, EventArgs e)
    {
        timer1.Start();
    }

    private void dataGridTable_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void DisplayText(object sender, EventArgs e)
    {
    }

    private void serialPort1_DataReceived
(object sender, System.IO.Ports.SerialDataReceivedEventArgs e)
    {
        RxString = serialPort1.ReadExisting();
        this.Invoke(new EventHandler(DisplayText));
    }

    private void button3_Click(object sender, EventArgs e)
    {
        serialPort1.Open();
        if (serialPort1.IsOpen)
        {
            MessageBox.Show("connected");
        }
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        #region imran's code

        #endregion

        String s = serialComm.receive();
        if (s.ToCharArray()[s.Length - 2] == 't')
        {
            s = s.Substring(0, s.Length - 2);
            dataGridTemp.Rows.Add(s, DateTime.Now.ToShortTimeString());
        }
    }

```



```

        }
    }

    private void dataGridView2_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void dataGridViewLight_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void dataGridViewHumid_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void dataGridViewTemp_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {
    }

    private void label12_Click(object sender, EventArgs e)
    {
    }

    private void panel1_Paint(object sender, PaintEventArgs e)
    {
    }
}

```